

Objectifs : Réaliser vos premières applications client-serveur utilisant le protocole TCP et comprendre le fonctionnement 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 Dialogue avec un serveur

Dans une application client-serveur, un serveur 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) et termine le traitement de ce client.

L'objectif de l'exercice est d'écrire le programme client. Ce dernier communiquera avec un serveur existant et s'exécutant sur une machine de la salle de TD/TP. La sortie standard du serveur sera projetée pendant la séance et permettra de vous guider dans la définition du protocole d'échange. L'adresse (IP + numéro de port) de la socket d'écoute vous sera fournie.

Ecrire et tester le programme client sachant que l'objectif à l'exécution est d'avoir un affichage correct des messages envoyés au serveur et un nombre minimum d'octets échangés (réduits à des données utiles).

2 Programmer le serveur

Ecrire le programme serveur de l'exercice précédent.

Ensuite, exécuter votre programme client et programme serveur sur des machines différentes et assurez vous du bon fonctionnement de votre application.

Que se passe-t-il si le client se termine avant l'envoi d'un message ?

3 Les échanges en TCP (Pendant le TP)

L'objectif de cet exercice est de mettre en évidence des propriétés du protocole TCP vues en cours.

Ecrire deux programmes :

- un programme client qui envoie successivement deux chaînes de caractères saisies au clavier (il fait donc 2 envois) et affiche le nombre total d'octets envoyés. La taille d'une chaîne de caractères saisie au clavier ne dépassera pas 32 caractères.
- un programme serveur qui reçoit une suite d'octets de taille maximum 124 et qui affiche le nombre d'octets reçus. Le serveur ne fait donc qu'une seule réception.

Exécutez les deux programmes (sur deux machines différentes) en s'assurant que les deux chaînes de caractères ont été toutes les deux envoyées avant que le serveur ne soit en réception. Qu'observez vous ? Discutez du résultat avec votre chargé(e) de TP et si nécessaire, corrigez votre programme.

Ensuite :

1. modifier le programme client pour qu'il envoie en boucle une même chaîne de caractères saisie au clavier ou lue depuis un fichier (taille maximum 2000 caractères). Le nombre de chaînes à envoyer (nombre d'itérations) sera un paramètre de votre programme. Enfin, le programme affichera le nombre total d'octets effectivement envoyés depuis la première émission (peut importe le type des messages). Afficher aussi le nombre d'octets supposés être envoyés.
2. modifier le programme serveur pour qu'il puisse recevoir les chaînes de caractères envoyées par le client. Le serveur affichera à chaque réception le nombre total d'octets effectivement reçus par le client depuis la première réception (peu importe le type des messages).
3. Exécuter les deux programmes (sur deux machines différentes !) en faisant varier la taille du message et le nombre de messages à envoyer de quelques uns à plusieurs milliers. Qu'observez vous ? Le nombre total d'octets envoyés est-il toujours égal au nombre d'octets reçus ? Si ce n'est pas le cas, expliquer le problème et corriger votre programme. Il est rappelé qu'il n'y a pas de perte de paquets en TCP ni de duplication à la réception par la couche application.
4. Relancer votre application de manière à remplir le buffer de réception du serveur et le buffer d'envoi du client. Que se passe-t-il ?

4 Optionnel mais recommandé : à faire chez soi et en groupe

Exécuter un client qui communique avec un serveur s'exécutant dans un autre sous-réseaux de l'internet a l'avantage de mieux mettre en évidence le comportement de TCP en comparaison avec une exécution au sein d'un réseau local.

Pour s'assurer de la robustesse de votre implémentation et corriger des éventuelles erreurs, vous êtes invités à tester l'application de l'exercice précédent de la manière suivante : demandez à un(e) camarade, habitant un autre quartier que le votre, de lancer un serveur puis exécutez le client depuis chez vous. Observez les résultats et s'ils ne sont pas cohérents avec le comportement attendu, corriger vos programmes.

Remarque : il sera probablement nécessaire de configurer votre box pour "ouvrir un port" avant de pouvoir réaliser une communication entre un client et un serveur. Vous trouverez des tutos sur internet en fonction de votre fournisseur d'accès.