

M4102C : Programmation répartie

TP1: Service de communication en mode datagramme (UDP)

Objectif :

Mise en œuvre de l'**API socket** du système **Linux**.

Travail demandé :

Développer en langage **C**, une applications répartie gérant une **communication unicast UDP**.

Spécifications :

a)- Programme **emetteur.c**

- l'adresse et le numéro de port destination seront passés en paramètres du main
- le nombre et la validité des paramètres seront contrôlés
- l'adresse de la socket distante sera configurée pour le mode unicast
- une suite de chaînes de caractères entrées au clavier, seront émises. Chaque chaîne sera préfixée par le nom de la machine, la date et l'heure
- le programme se terminera quand la chaîne "**stop**" sera saisie

b)- Programme **recepteur.c**

- la socket locale sera configurée pour le mode unicast
- le port dynamique sera affiché
- la suite de chaînes de caractères reçues sera affichée à l'écran
- le programme sera arrêté avec "Ctrl-C" quand la chaîne "**stop**" sera reçue

Exercice1 :

Sur le poste de travail, avec **gedit**, coder les 2 programmes **emetteur.c** et **recepteur.c**.
Avec **gcc**, générer les programmes exécutables **emetteur** et **recepteur**.

Ouvrir une fenêtre terminal et lancer le programme **recepteur**.
Ouvrir une fenêtre terminal et lancer le programme **emetteur** en utilisant l'interface "**loopback**" (@IP 127.0.0.1) et le port affiché par le programme **recepteur**.
Mettre au point les 2 programmes jusqu'à ce que l'application répartie fonctionne correctement.

Exercice2 :

Note :

Demander à l'enseignant de faire créer par l'administrateur système du département, sur chaque poste de travail, une interface spéciale appelée "**tap0**" à l'@IP **192.168.1.1** qui permet à la machine hôte de communiquer avec une machines virtuelle UML sur son interface **eth0**.
Vérifier avec la commande "**/sbin/ifconfig -a**" que cette interface est présente sur la machine hôte.

Créer une machine UML **ST1** avec la commande **uml_lancer_serveur**.
Configurer l'interface **eth0** pour que cette machine communique avec la machine hôte :

- **ifconfig eth0 192.168.1.10/24**
- **route add default gw 192.168.1.1**

Sur la machine **ST1**, sous le compte **root**, créer le répertoire **P112-TP1**.
Sur la machine hôte, télécharger (cmde **scp**) les programmes exécutables **emetteur** et **recepteur** dans le répertoire **P112-TP1** de la machine **ST1**.

Créer un réseau virtuel Ethernet **réseau#1** avec la commande **uml_creer_reseau**.
Créer une machine UML **ST2** avec la commande **uml_lancer_machine**.

Connecter la machine **ST1** au **réseau#1** par son interface **eth1** à l'@IP **10.0.0.1/24**.
Connecter la machine **ST2** au **réseau#1** par son interface **eth1** à l'@IP **10.0.0.2/24**.
Vérifier que ces machine communiquent.

Sur la machine **ST2**, sous le compte **root**, créer le répertoire **P112-TP1**.
Sur la machine **ST1**, télécharger (cmde **scp**) le programme exécutable **recepteur** dans le répertoire **P112-TP1** de la machine **ST2**.

Sur la machine **ST2** lancer le programme exécutable "**recepteur**".
Sur la machine **ST1** lancer le programme exécutable "**emetteur**".

Vérifier que l'application répartie fonctionne correctement.

Arrêter les programmes sur chaque machine virtuelle.
Arrêter les 2 machines (cmde **halt**)