Master Sciences & Technologies
Mention Informatique
Université Bordeaux 1

Architecture des réseaux
et systèmes associés
Travaux Dirigés

Le grand mélange

sockets, fork, signaux, *etc.*

Il s'agit de développer le service `my_inet`, version allégée du *super serveur* `inet` . La tâche d'inet est de surveiller un ensemble de ports correspondant à un ensemble de services. Lorsqu'une requête arrive sur un des ports surveillés, `my_inet` lance le service concerné avec la socket appropriée en entrée et sortie standards. De ce fait les serveurs doivent être conçus pour être (aussi) lancé avec inet.

À l'instar du fichier de configuration `/etc/inetd.conf` un tableau décrira l'ensemble des services à rendre ; pour chaque service, ce tableau contiendra au moins son nom, son port d'attache, son protocole, le chemin absolu de son programme, ses paramètres et un champs WAIT désignant le fait qu'il faille attendre ou non la fin du processus lancé pour se charger à nouveau du port concerné. Concrètement, en mode WAIT on lance un serveur – en général un démon qui va rendre le service et traiter un certain nombre de requêtes – tandis qu'en mode NOWAIT on lance un processus traitant uniquement la requête en attente.

On prendra le temps de lire le paragraphe *Configuring Inetd* de l'aide en ligne de la libc, on pourra aussi se renseigner sur le programme `xinet`.

Étape 1. Programmer le service TCP `echo` qui retourne les données qu'on lui transmet. Tester ce service. Créer ce même service en mode UDP.

Étape 2. On supposera (aujourd'hui) qu'on a uniquement des services TCP en mode wait :

1. Écrire le super-serveur en supposant que les processus lancés ne se terminent pas (proposer un mode debug et un mode log) ;
2. adapter votre service echo au super-serveur ;
3. tester votre super-serveur en créant plusieurs serveurs echo ;
4. créer le programme `lance nb executable` qui lance `nb` exécutions du programme `executable` en parallèle tout en affichant périodiquement la liste des `pid` des processus lancés qui sont toujours en exécution ;
5. modifier votre super-serveur pour tenir compte de la terminaison des serveurs (en utilisant le code créé question précédente) ;
6. tester votre solution en tuant les serveurs echo.

....