

### **Petit exercice Perl :**

Ecrire deux programmes en Perl.

Le premier que nous appellerons « client » établit  $n$  connexions TCP à un serveur.

Sur chaque connexion, il envoie toutes les 10ms un timestamp en millisecondes.

Le second que nous appellerons « serveur » réceptionne les timestamps provenant des  $n$  connexions, les ordonne et les range dans un fichier.

Ces différentes actions sont faites en continue, le client envoie des données en permanence, le serveur les reçoit en continu et, au fur et à mesure les ordonne et les stocke.

Ces deux programmes doivent être écrits en Perl, on privilégiera les modules simples et légers si besoin.

Plusieurs points :

- Lorsque  $n$  grossit, le même timestamp risque d'être envoyé plusieurs fois. Nous voudrions que dans ce cas les timestamps reçus n'apparaissent dans le fichier qu'une seule fois.
- faire attention au cas où un client tourne sur une machine différente avec un décalage horaire.
- Nous voudrions que le fichier soit complété le plus régulièrement possible.
- La solution doit être optimisée au mieux.
- La solution doit être lisible, un minimum documentée et testée.
- La solution sera envoyée sous forme d'une archive (.tar.gz par exemple).

Un exemple de résultat possible :

```
1737030375.988
1737030375.990
1737030375.991
1737030375.992
1737030375.993
1737030375.994
1737030375.995
1737030375.996
1737030375.997
1737030375.998
1737030375.999
1737030376.001
1737030376.002
1737030376.003
1737030376.004
1737030376.005
1737030376.006
1737030376.009
1737030376.010
1737030376.011
1737030376.012
...
```