

# PROJET - PROGRAMMATION RÉSEAU ET SYSTÈME

4TC PRS - INSA Lyon

## Le meilleur serveur pour un client donné

L'objectif du projet est de réaliser le design et l'implantation de serveurs adaptés à trois scénarios proposés :

- Le **Scénario 1** utilise un client de type *client1*.
- Le **Scénario 2** utilise un client de type *client2*.
- Le **Scénario 3** utilise plusieurs clients de type *client1*.

Dans tous les scénarios, la métrique utilisée pour mesurer les performances des serveurs sera le débit des serveurs.

### Rendu

Vous devez rendre sur moodle une archive nommée *NomDuGroupe.tar.gz*, où *NomDuGroupe* représentera le nom que vous allez choisir pour votre groupe. L'archive doit contenir trois répertoires : *./bin*, *./src* et *./pres*.

Le dossier *./bin* contiendra trois fichiers exécutables nommés *serveurX-NomDuGroupe*, avec *X* étant le numéro du scénario. Attention, les trois fichiers doivent exister même si vous proposez un seul serveur pour les trois scénarios. Les trois serveurs doivent être exécutables sur les machines du département et ils doivent pouvoir être lancés avec la commande :

```
> ./serveurX-NomDuGroupe NuméroPort
```

où *NuméroPort* représente le numéro du port public sur lequel votre serveur va écouter.

Le dossier *./src* doit contenir toutes les fichiers source de votre projet, et un fichier *Makefile* générant les trois exécutables décrits ci-dessus.

Le dossier *./pres* contiendra votre support de présentation de projet, en format pdf.

### Dates

- Avant le 19.11, 18h : vous devez envoyer un mail à l'adresse *razvan.stanica@insa-lyon.fr*, avec le titre "Groupe Projet PRS" et contenant le nom et prénom des étudiants faisant partie du groupe de projet et le nom du groupe, utilisé ensuite dans le rendu. Le groupe de projet peut être constitué de deux étudiants maximum (et un minimum). Le nom du groupe peut contenir des chiffres et des lettres, sans espaces et/ou caractères spéciaux.
- Avant le 15.12, 12h : vous devez déposer votre rendu de projet sur moodle.
- 15.12, 8h-12h : présentation du projet. Des instructions concernant la présentation se trouvent à la fin de ce document.

**Le non-respect de chacune de ces dates entraine un malus de 10% sur la note finale.**

## Fonctionnement des clients

Comprendre le fonctionnement des clients et les raisons derrière représente un des objectifs du projet. Les différents clients ont des comportements différents. Cependant, quelques éléments communs peuvent vous être utiles :

1. Le client utilise des sockets UDP pour transférer un fichier depuis le serveur.
2. Le client commence par la transmission d'un message contenant la chaîne de caractères "SYN" sur le port public du serveur.
3. Le client attend le message "SYN-ACKPPPP" de la part du serveur, où "PPPP" représente le numéro d'un nouveau port, utilisé pour l'échange de données (le numéro de port doit être entre 1000 et 9999).
4. Le client répond avec un message "ACK" sur le port public.
5. Juste après la fin de la connexion, le client envoie un message contenant le nom du fichier recherché au serveur, sur le port communiqué par celui-ci dans la phase de connexion.
6. Le client attend des messages de données qui commencent avec un numéro de séquence, en format chaîne de caractères, sur 6 octets.
7. Quand le message est bien reçu, le client répond avec un message "ACKSSSSSS", où "SSSSSS" est le numéro de séquence du dernier segment reçu en contigu. Ce message est transmis sur le port dédié au client.
8. A la fin de la transmission du fichier, le client attend un message "FIN" de la part du serveur.

Pour lancer le client, vous devez utiliser la commande :

```
> ./clientX IPServeur NuméroPortServeur NomFichier
```

Le client se lancera en mode *debug*, en affichant certains messages qui vous permettront de comprendre son fonctionnement. Si vous voulez lancer le client sans ces messages, la commande à utiliser est :

```
> ./clientX IPServeur NuméroPortServeur NomFichier 0
```

## Présentation

Quelques conseils pour la présentation :

- vous auriez 10 minutes de présentation et 5 minutes de questions. Le respect du timing sera un critère très important dans l'évaluation de votre présentation ;
- vous pouvez vous aider de diapos, dans quel cas vos diapos en format .pdf doivent être déposés sur moodle avant la présentation (voir instructions ci-dessus) ;
- l'objectif est d'expliquer vos choix, si bien au niveau conception qu'au niveau implantation ;
- dans la plupart des cas, il serait probablement judicieux d'aborder chaque scénario à part ;
- il faut justifier les valeurs que vous avez choisies pour les différents paramètres utilisés par vos serveurs dans le contrôle de congestion ;
- des petites courbes d'évaluation de performances seront appréciées ;
- ne faites pas une lecture de votre code, mais expliquez les problèmes rencontrés et les solutions trouvées ;

- n'hésitez pas à décrire des choses que vous auriez voulu faire, mais vous n'avez pas pu, par manque de temps ou de compétences.

## **Planning des présentations**

### **Romain Pujol - TD C**

- 8h30 : KoolAndTheGang
- 8h45 : PetitsRennesSavants
- 9h00 : MISDUC
- 9h15 : ClientAbuser
- 9h30 : Manjikai
- 10h15 : pln
- 10h30 : Nestor
- 10h45 : SeTTooOOOoODUuuuUUr
- 11h00 : 233
- 11h15 : Julia

### **Frédéric Le Mouel - TD D**

- 8h30 : lePic
- 8h45 : lezgo
- 9h00 : LosOptimizadores
- 9h15 : LesSemiCroustillants
- 9h30 : LesCopaings
- 10h15 : n0\_c0nsensus
- 10h30 : DetecteurML
- 10h45 : SSHF
- 11h00 : SegmentationFault
- 11h15 : cless

### **Razvan Stanica - TD E**

- 8h30 : PeRSeverance
- 8h45 : COM\_SUCKET
- 9h00 : ElYouDP
- 9h15 : SP33DRUNN3RS
- 9h30 : LesTryhardeusesDuDimanche
- 10h15 : Ableton
- 10h30 : TCPasFacile
- 10h45 : TeamXY
- 11h00 : HanSynack
- 11h15 : Seagull