TP Messagerie

Compte rendu

Binôme 4IF Groupe 2 :

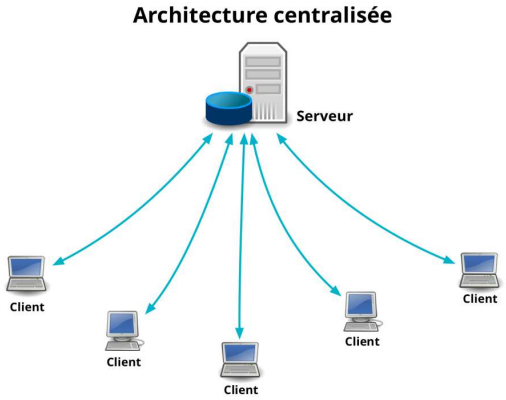
FLANDRE Corentin

THOMAS Colin

Sommaire

I. Contexte de l'application

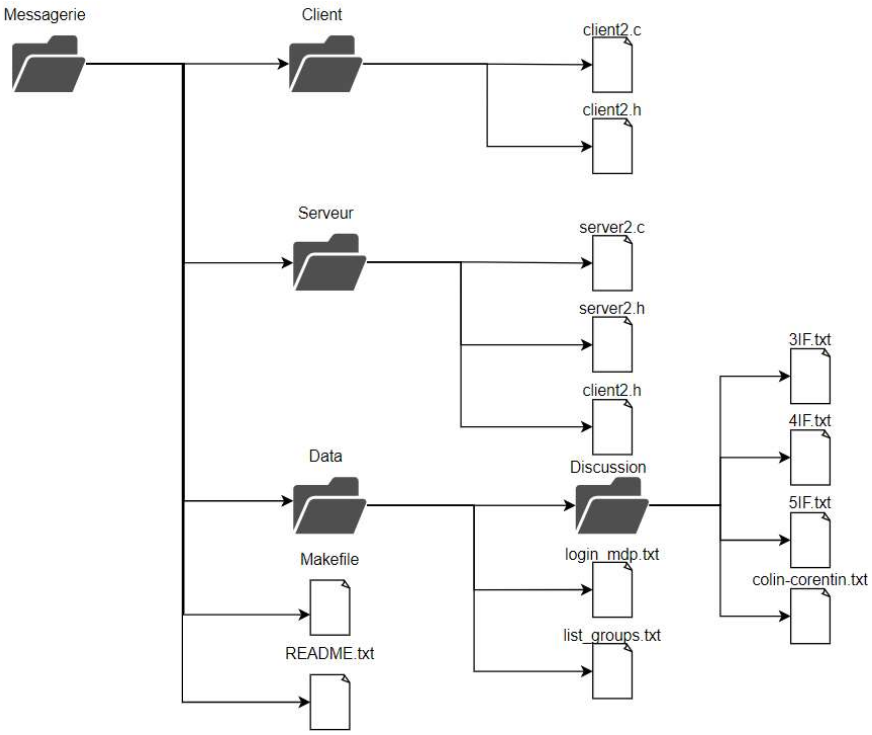
Dans le cadre de la matière de Programmation Réseau, l'objectif du TP Messagerie est de créer une application de messagerie mettant en place une architecture centralisée. Ce TP permet donc de manipuler des Sockets en C.



II. Architecture

Pour ce TP, nous avons utilisé le socle architectural de base disponible sur moodle permettant à des clients de tous s'envoyer des messages à l'aide d'un serveur. Les points importants de la modification de cette architecture sont:

- Modifications des classes pour ajouter les fonctionnalités
- Ajout d'un dossier Data pour stocker des informations en clairs (liste des groupes, login/mdp, historique des groupes)
- Ajout d'un fichier Makefile pour faciliter grandement la compilation et l'exécution



III. Fonctionnalités

Nous avons implémenté/modifié différentes fonctionnalités pour notre messagerie:

- Connexion d'un client au serveur avec login et mot de passe (qui sont stockés niveau serveur)
- Ajout d'un nouveau client à l'application de messagerie (il s'agit du même principe que la connexion, le compte de l'utilisateur sera créé si il c'est la première fois qu'il apparaît et son mot de passe sera celui utilisé dans la commande)
- Connexion de plusieurs clients en simultanés
- Création d'un groupe (avec nom de groupes et ajout de clients)
- Ajout d'un ami (il s'agit du même principe que la création de groupes, on considère que discuter est équivalent à parler à un groupe avec deux personnes (destinataire et receveur))
- Discussion en direct de deux clients connecté au même groupe
- Sauvegarde de l'historique des messages pour chaque groupe
- Chargement de l'historique des messages du groupe lors de la venue
- Lister les groupes et amis dont l'utilisateur a accès
- Déconnexion d'un client du serveur
- Ajout de l'horaire du message lors de l'envoi d'un message (et sauvegarde dans l'historique)

IV. Manuel d'utilisation et exemple d'utilisation

IV.1. Manuel d'utilisation

Compiler le projet :

> make

Supprimer les fichiers compilés :

> make clean

Depuis le fichier racine du projet (Messagerie) pour lancer le serveur:

> make exec_serveur

Avec valgrind :

> make exec_serveur_valgrind

Depuis le fichier racine du projet pour lancer un client:

> ./Client/client 127.0.0.1 "login" "password"

Depuis le client connecté:

- lister les groupes auxquels le client participe :
  > list
  - choisir ensuite un groupe à consulter/contacter :
    > "nom_du_groupe"
    - envoyer un message à tous les membres du groupe :
      > "votre_message"

- revenir à l'écran d'accueil :
  > home

- créer un groupe :
  > create
  > “nom_du_groupe”
  > “nom_du_membre” %répeter autant de fois que de membres (il n’est pas nécessaire de se spécifier soi-même)
  > finish

- se déconnecter :
  > quit

IV.2. Exemple d’utilisation

Nous avons inséré des données afin d’avoir un scénario d’utilisation fonctionnel pour exemple. Celui-ci utilise un serveur et deux clients avec les login colin et corentin (les mots de passes de ces utilisateurs déjà existants sont 1234

Attention il faut éviter de copier/coller les commandes de l’exemple à cause des caractères invisibles, il est préférable de les recopier

Serveur
make && make exec_server

Client numéro 2	Client numéro 1
$ ./Client/client 127.0.0.1 corentin 1234	$ ./Client/client 127.0.0.1 colin 1234
>list	>list
>3IF	>3IF
>Tant que je n’écris pas home ni quit je peux ecrire des messages	
>home	>home
>create	>quit
>groupepromo2024	
>colin	
>alexis	

>finish	
>list	
>quit	