Dossier de programmation

Post Office Protocol 3

Polytech Lyon

Jérémy DOLLE – Mélanie PETITCUENOT

Sommaire

Ce document présente l’application client/serveur conforme à la norme RCF 1939 « Post Office Protocol 3 ». Vous y trouverez les différentes fonctionnalités de l’application ainsi que les éventuelles erreurs que vous pourrez rencontrer lors de son utilisation.

[I. Application Serveur 2](#_Toc478392720)

[II. Application Client 4](#_Toc478392721)

[a. Fonctionnalités 4](#_Toc478392722)

[b. Erreurs possibles 6](#_Toc478392723)

[III. Connexion sécurisée 7](#_Toc478392724)

# Application Serveur

L’application serveur est très facile d’utilisation et l’interface est simple. Pour le lancer, il suffit d’appuyer sur le bouton. Le serveur passe alors en mode écoute et attend une connexion TCP de la part d’un client.

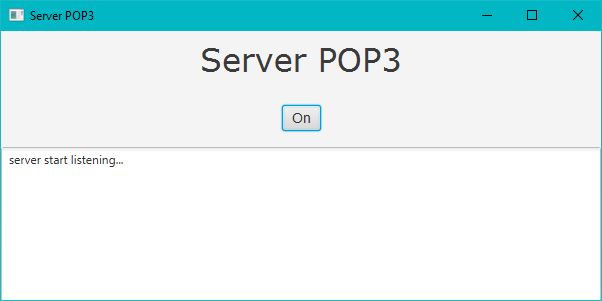


Figure : Fenêtre de lancement du serveur:

Dès qu’un client veut joindre le serveur, une connexion TCP est automatiquement établie. Par la suite, le client peut envoyer différentes commandes. Cependant il doit commencer par s’identifier avec un nom d’utilisateur et un mot de passe en utilisant la commande APOP.

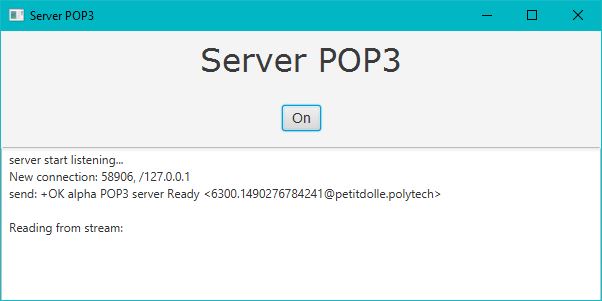


Figure : Connexion TCP établie avec un client, en attente de l’authentification

Seuls les clients ayant un identifiant et un mot de passe enregistrés dans le fichier utilisateurs du serveur peuvent se connecter et avoir accès aux différentes fonctionnalités. Chaque utilisateur possède un fichier JSON dans lequel tous les messages qu’il a reçus sont enregistrés.

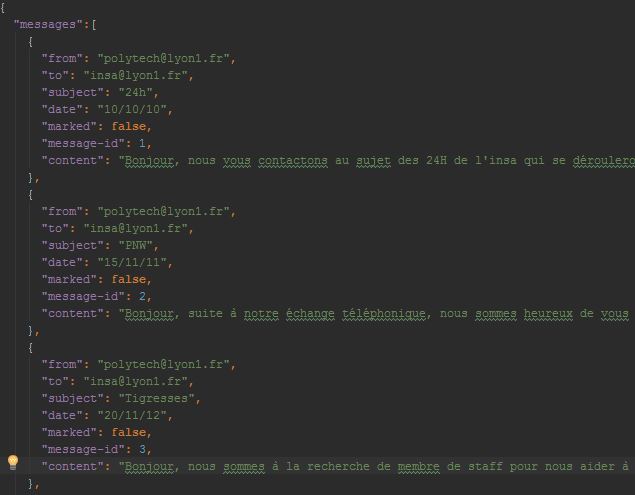


Figure : Fichier utilisateur

# Application Client

### Fonctionnalités

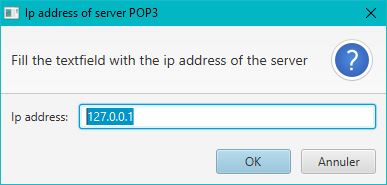
 Lors du lancement de l’application, le client commence par renseigner l’adresse IP du serveur qu’il souhaite joindre, le port étant renseigné directement par l’application. Suite à cela, l’application cliente se connecte au serveur via une connexion TCP.

Figure : Définition de l'adresse IP du serveur

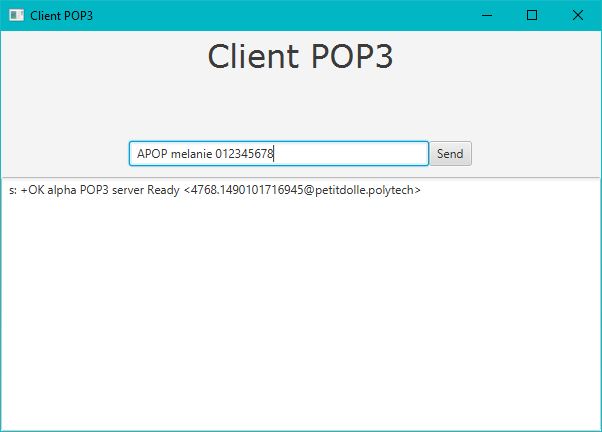
 Le client doit ensuite se connecter en utilisant la commande APOP et en renseignant son identifiant et son mot de passe. L’utilisation de toutes autres commandes sera refusée par le serveur et retournera une erreur. De plus, si le client échoue plus de trois fois à se connecter, la connexion TCP est coupé par le serveur et le client doit relancer l’application pour retenter de se connecter.

Figure : Connexion TCP établie avec le serveur, lancement de la commande pour s'authentifier

Une fois connecté, le client a à sa disposition plusieurs commandes afin d’obtenir des informations sur ses mails :

* Supprimer (DELE) + numéro d’un message : permet de marquer un message qui sera supprimer lors de la déconnexion ;
* Lister (LIST) : renvoie le nombre de messages avec la taille totale et une liste avec le numéro de chaque message (non-marqué pour être supprimé) et sa taille en octet ;
* Quitter (QUIT) : supprime les messages marqués puis déconnecte le client et ferme la connexion TCP ;
* Lecture (RETR + numéro d’un message) : renvoie le détail du message ciblé avec l’expéditeur, le receveur, le sujet, la date et le corps du message ;
* Statistique (STAT) : renvoie le nombre de message sur la boite mail ainsi que l’espace occupé en octet.

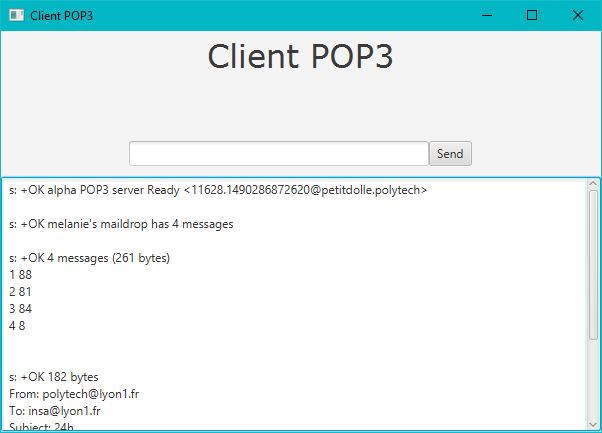


Figure : Exemple d'échange du point de vue du serveur

Figure : Exemple d'échange du point de vue du client

### Erreurs possibles

Le client peut rencontrer différentes erreurs durant l’utilisation de sa boite mail. Les erreurs lui seront retournés et explicités. Voici les cas les plus fréquents.

**Echec de connexion :**

Au bout de trois essais infructueux de tentative de connexion, le client est déconnecté. Les échecs des tentatives peuvent être dues à une erreur dans l’écriture de l’identifiant ou du mot de passe.

**Numéro de message invalide :**

Les commandes *Supprimer* et *Lecture* oblige le client à préciser le message auquel il souhaite les appliquer. Si le client ne renseigne pas le numéro d’un message ou en renseigne un qui n’existe pas, une erreur lui sera renvoyé.

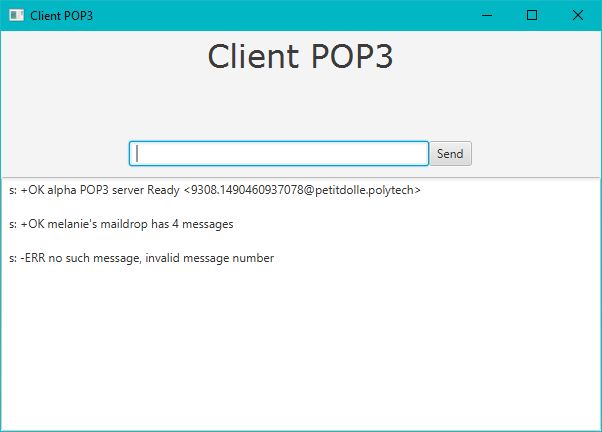


Figure : Exemple d'erreur côté client, le message demandé n'existe pas

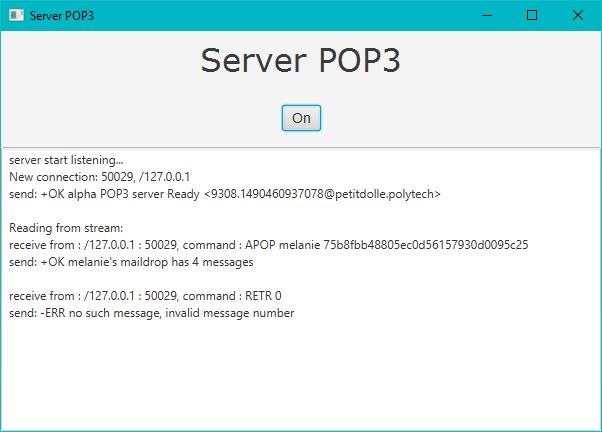


Figure : Exemple d'erreur côté serveur, le message demandé n'existe pas

# Connexion sécurisée

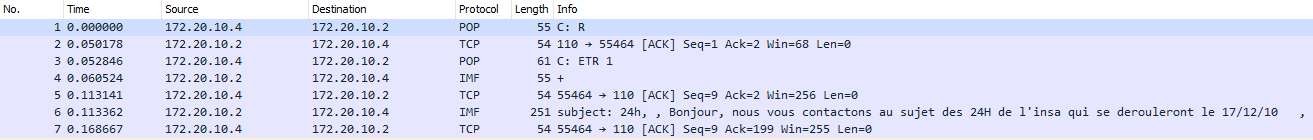
 La connexion entre le client et le serveur est sécurisé en utilisant le protocole POP3S. Alors, le mot de passe de l’utilisateur est chiffré lors de son envoie au serveur, tout comme les mails en provenance du serveur. Ainsi, un éventuel pirate ne peut pas intercepter les mots de passes pour usurper l’identité d’un utilisateur ou lire les messages qui transitent.

Figure : Échange non-sécurisé

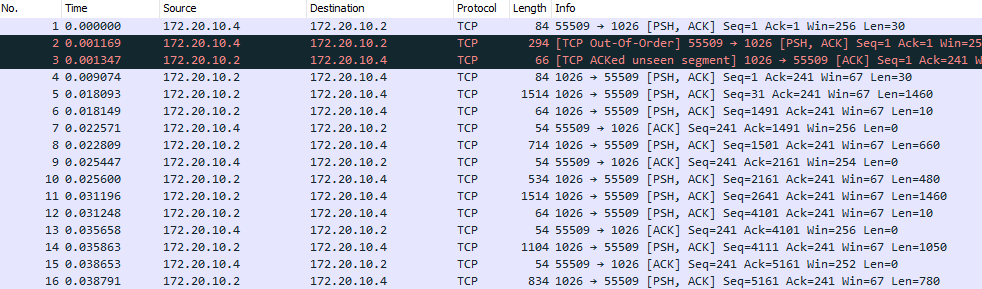


Figure : Échange sécurisé