

Année universitaire	2017/2018	Année	4A
Département	Informatique		
Matière	IPC		
Enseignant	C.GERTOSIO		
Intitulé TD/TP :	TP Client-Serveur « Post Office Protocol 3 »		

Le TP sera noté en trois parties: graphes et table, démonstration du logiciel et rapport écrit.

Il s'agit de concevoir et réaliser une application client/serveur conforme à la norme RFC 1939 « Post Office Protocol3 ».

1. On utilisera la méthode des automates d'états finis à prédicats pour effectuer l'analyse et la programmation de cette application. On vous demande de réaliser :

- a. le graphe et la table de transitions du serveur POP3. Vous respecterez les états définis par la norme RFC 1939.

Pour faire le graphe, vous pourrez vous aider d'un ou plusieurs scenarios.

2. On voudrait implanter les commandes suivantes du client-serveur POP3 APOP, STAT, RETR et QUIT.

- a. Créer le programme du serveur POP3.

-Vous veillerez à ce que la programmation soit conforme aux automates (une commande ne sera exécutée que dans le bon état). Vous utiliserez les programmes Telnet ou Putty pour simuler le client

- b. Créer le programme du client POP3

-L'application client permettra à l'utilisateur de relever le courrier(Elle ne permettra pas à l'utilisateur des taper les commandes). Les messages relevés sur le serveur POP3 seront rangés dans un fichier sur le client et affichés à l'écran.

Vous utiliserez les classes socket de Java pour programmer client et serveur. Une démonstration vous sera demandée. Pour cette démonstration, la boîte à courrier d'un utilisateur devra contenir au moins 3 messages au format RFC 5322.

3. Le dossier de programmation comprendra une notice explicative du fonctionnement des programmes client et serveur ainsi que les algorithmes des parties client et serveur.

Annexe

Gestion du courrier

On va simuler les dépôts de courriers du serveur. Pour cela, on utilisera un fichier pour chaque User. Ce fichier contiendra au moins 3 messages conformes à la norme RFC 5322.

La norme RFC 5322 donne le format de tous types de messages. Un message simple est composé d'une enveloppe et d'un corps:

```
Header <CR><LF>
Header <CR><LF>
Header <CR><LF>
<CR><LF>
Ligne <CR><LF>
Ligne <CR><LF>
.<CR><LF>
```

Ligne a une taille inférieure à 998 caractères terminée par les 2 caractères <CR><LF>. Dans la pratique, la taille est inférieure ou égale à 78 caractères terminée par les 2 caractères <CR><LF>.

Un message se termine par un point en début de ligne suivi des 2 caractères <CR><LF>.

<CR> représente 1 octet contenant la valeur 13 ou 0dh. Ce caractère, envoyé sur un écran, renvoie le curseur en début de ligne.

<LF> symbolise 1 octet contenant la valeur 10 ou 0Ah. Ce caractère, envoyé à un écran, fait passer le curseur à la ligne suivante.

Exemple de message

```
From: John Doe <jdoe@machine.example>
To: Mary Smith <mary@example.net>
Subject: Saying Hello
Date: Fri, 21 Nov 1997 09:55:06 -0600
Message-ID: <1234@local.machine.example>
```

This is a message just to say hello.
So, "Hello".

Paramétrage du logiciel client

On suppose que le logiciel client relève le courrier sur le serveur et qu'il est paramétré pour ne pas effacer les messages lus (option: laisser les messages sur le serveur).

Le contrôle d'authentification

Lorsque le logiciel client envoie la commande APOP au serveur, celui-ci fait un contrôle MD5. Dans ce TP, vous ne sécuriserez pas la commande APOP, cela sera réalisé dans un deuxième temps.

Les liens utiles

La traduction française de POP3: <http://abedrfc.free.fr/rfc-vf/rfc1939.html>

La traduction française de RFC 5322: <http://abedrfc.free.fr/rfc-vf/pdf/rfc5322.pdf>