

TP - LIF12 système d'exploitation

Projet client/serveur de mail

mars-avril 2017

I Introduction

Vous devez programmer deux logiciels :

- Un **client mail**, c'est à dire un logiciel capable de contacter un serveur pour lui demander d'envoyer un mail.
- Un **agent de transfert de mails (MTA)** ou serveur mail qui permet de réceptionner les demandes de client. Il délivre (ou sauvegarde) les mails envoyés à un utilisateur local et transfert à d'autres serveurs les mails destinés à un domaine extérieur.

Les programmes proposés devront être compatibles avec les serveurs mails existant. Nous vous fournirons notamment un serveur de test basé sur un de ces logiciels (postfix) et nous vous inviterons à tester vos codes avec les serveurs de vos fournisseurs d'accès internet ¹

Dans ce projet, il y a plusieurs niveaux de difficulté possible. La version de base (décrite plus loin) vous permettra d'obtenir une note proche de la moyenne. Les versions améliorées vous permettront d'augmenter cette note.

II Description du client

Vous devez faire un client qui prend plusieurs paramètres, 2 d'entre eux sont obligatoires : le *serveur à contacter* et *son port*. Les autres paramètres sont optionnels :

- le ou les mails destinataires,
- le mail de celui qui envoie,
- le titre du mail,
- le texte du mail,
- ...

Pour simplifier les tests, vous devez permettre à votre programme d'envoyer un mail sans poser de questions. Cela signifie que si les informations nécessaires sont données en option, le mail est envoyé directement. Ces informations sont : le titre, le *from*, le *to* et le texte du mail. Si ces informations ne sont pas fournies, votre programme peut proposer une interface interactive qui permet de compléter les champs manquants.

II.1 Amélioration du client

Plusieurs améliorations sont possibles pour cette partie :

- ☺ envoyer des mails à un groupe de personnes ;
- ☺ faire une interface particulièrement bien conçue et facile d'utilisation ;
- ☺ gérer les pièces jointes ;
- ☺ gérer le SSL/TLS et l'authentification.

III Agent de Transfert de Mails

Ce programme doit servir de serveur de mails. C'est à dire qu'il doit pouvoir attendre des clients, lire la requête d'un client et traiter son mail. Ce traitement peut être de délivrer le

1. Attention, ils ont tendance à cataloguer des machines comme émettrices de spam. De plus un bon nombre demande l'authentification de l'utilisateur qui envoie le mail. Cela est déconseillé sur une connexion non chiffrée.

mail localement ou de le transmettre à un autre serveur. Dans la version de base, ce serveur fonctionnera sur le mode *relay host*, c'est à dire que tous les mails sont transmis au même serveur (celui du fournisseur d'accès) sauf ceux qu'il délivre localement.

Le programme doit donc avoir 2 paramètres obligatoires : le *port d'écoute*, le *domaine local*², la liste des utilisateurs de ce domaine³. Il doit aussi avoir plusieurs paramètres optionnels :

- le *relay host* c'est à dire le nom et le port du serveur à contacter pour transmettre les mails, la présence de ce paramètre entraîne le fonctionnement du serveur sur le mode *relay host*;
- les *alias* c'est à dire les substitutions d'adresses à utiliser (lors de l'implémentation de l'amélioration correspondante);
- ...

III.1 Amélioration

La liste non exhaustive des améliorations possibles :

- ☹️ une gestion particulièrement robuste des erreurs du client⁴;
- 😊 traiter les *alias* ou utiliser des filtres de recherche et remplacement;
- ☹️ implémentation d'une file d'attente des mails à envoyer dans le cas où les serveurs ne sont pas disponibles;
- 😊 parallélisation de la gestion des clients;
- 😊 utilisation d'un logiciel d'analyse comme `spamassassin` pour détecter les spams;
- 😊😊 fonctionnement en mode standard, le serveur doit pouvoir trouver et contacter directement le serveur de destination.
- ...

IV Notation

C'est votre enseignant de TP qui notera votre projet, cela signifie que c'est à lui de vous expliquer les détails de ce qui est demandé. De plus, vous ne devez pas changer de groupe.

Le barème prévu est le suivant :

- 5 points pour un client de base;
- 5 points pour un serveur de base;
- +3 ou -6 points en fonction de critères annexes :
 - votre comportement en TP (facilité de compréhension, motivation, ...);
 - la qualité du code (robustesse du programme, clarté du code, originalité de la méthode...);
 - qualité du rendu (ensemble de codes qui compile, qui fonctionne sans bug, avec un mode d'emploi pour l'utiliser ...).

Pour augmenter la note, il faut implémenter des améliorations. Le nombre de points gagné dépend de la difficulté de cette dernière (attention, pour le moment ce n'est qu'une estimation et le montant des points peut varier) :

- ☹️ $\simeq 1$ pt : amélioration simple mais utile .
- 😊 $\simeq 3$ pt : amélioration compliquée demandant un travail conséquent et une bonne compréhension de nouvelles notions⁵.
- 😊😊 $\simeq 5$ pt amélioration difficile qui demande des actions que je ne connais pas encore ou la modification en profondeur des bibliothèques fournies⁶.

2. Le domaine de mail considéré comme local, c'est à dire l'adresse `@nom.de.domain.fr` pour laquelle ce serveur délivre directement les mails

3. Tout mail local destiné à un utilisateur qui n'est pas dans la liste sera refusé avec un code d'erreur approprié.

4. une gestion particulièrement mauvaise est considérée comme un bug et baissera la note de base.

5. Par exemple la gestion de pièce jointe demande la compréhension de l'utilisation des séparateurs dans les mails, des types MIME et de l'encodage base64.

6. Par exemple pour trouver le serveur de destination finale, il faut interroger le DNS pour obtenir cette information (ce que je ne sais pas faire personnellement) et avoir un client capable de parler à un nombre important de serveurs différents. Pour passer en mode SSL/TLS, il faut changer les fonctions de lecture/écriture de la bibliothèque `socklib`

DÉPARTMENT D'INFORMATIQUE

La date prévue pour rendre le projet est le mercredi 26/04 à 9h du matin via la case ProjetMail de sur tomuss. Vous devez rendre une archive avec un programme qui fonctionne, les sources pour le compiler ainsi qu'un makefile (ou n'importe quel système de compilation du même genre). De plus, il est conseillé de décrire ce que vous avez fait dans un fichier accompagnant l'archive afin que votre encadrant n'oublie pas certaines options. Enfin, nous devons vous voir travailler et avancer sur le projet, un programme de bonne qualité « sortie du chapeau » n'aura pas de note positive. De plus, vous devez être présents durant les séances : sur les 4 séances prévues, seule une absence est tolérée. Une absence supplémentaire vous donnera la note de 0 si elle n'est pas justifiée (et annulera la note de projet si toutes les absences le sont).