

Cours n°6

La messagerie



Sommaire

1. Envoi d'un message

- protocole SMTP
- Programmation réseau

2. Réception d'un message

- Protocole POP3
- Programmation réseau

Request For Comments (RFC)

Ensemble de documents spécifiant les protocoles liés à Internet,
Référence de la Communauté Internet (www.ietf.org/rfc)

Traduction en français (<http://abcdrfc.free.fr/>)

**J. Klensin, Simple Mail Transfer Protocol, RFC 2821,
2001**

**P. Resnick, Internet Message Format, RFC 2822,
2001**

J. Myers, Post Office Protocol - Version 3, RFC 1939, 1996

**M. Crispin, Internet Message Access Protocol – Version
4rev1, RFC 2060, 1996**

Agents de la messagerie

Trois agents (ou programmes conversationnels)

Communication inter-agents par l'intermédiaire de Sockets (TCP principalement)

Mail User Agent (MUA)

Client de messagerie (Thunderbird, Kmail, Outlook, ...)

Création de message par un l'utilisateur et envoi

Réception d'un message et lecture par un utilisateur

Mail Transfert Agent (MTA)

Acheminement des messages (courrier) de l'expéditeur au serveur du destinataire.

Mail Delivery Agent (MDA)

Distribution du courrier dans les boîtes des utilisateurs.

Application de filtres (anti-virus, anti-spam...)

Gestion des nœuds de la messagerie

Identification de la provenance du message

Commerce électronique,
Détection de la provenance des spams,
Services de sécurité

Gestion des nœuds MTA et MDA par des entités connues

Service informatique des entreprises,
Fournisseurs internet,
Pas de nœuds MTA et MDA sur des ordinateurs personnels

Envoi d'un message de l'utilisateur 1 à l'utilisateur 2

MUA1 > MTA1 > ... > MTAx > ... > MTAy > MDA > MUA2

Obligation d'utiliser le MTA (ou le webmail) de son organisme

Description

Format d'un message texte (RFC 2822)

Messages composés de caractères US-ASCII de 1 à 127,

Division en lignes

998 caractères au plus terminées par CR et LF

Champs d'entête

caractères imprimables de 33 à 126, suivies d'un « : »

exemple « subject: »

Corps de message (optionnel),

Code de fin de message (une ligne avec le caractère « . »)

Format d'un message non-texte (RFC 2049)

Transcodage par les procédures MIME en US-ASCII

Principes

Simple Mail Transfer Protocol

Protocole défini en 1982 (J.B. Postel) dans la RFC 821

Ouverture d'une connexion TCP bi-directionnelle sur le port 25

Suite de dialogues réseaux entre le MUA et le MTA

Dialogue (échange de requêtes / réponses)

Requête (par le MUA uniquement)

3/4 lettres + arguments

Réponse (par le MTA uniquement)

Code à 3 chiffres + arguments / message

Quatre types de nœuds SMTP

Originator Machine émettant le message

Relay Hôte intermédiaire (pas de modification possible du message)

Gateway hôte intermédiaire (modification possible du message)

Delivery Machine recevant le message

Syntaxe des requêtes

Mots de commande

HELO, EHLO adresse internet ou nom du domaine du MUA

Identification du MUA auprès du MTA

MAIL FROM : adresse de messagerie de l'expéditeur

1^{ère} étape de l'envoi d'un message

RCPT TO: adresse de messagerie d'un destinataire

Requête faite pour chaque destinataire

DATA message terminé par le code de fin de message

Envoi du message

RSET Annulation de l'envoi du message

VERFY Vérification de l'existence d'un destinataire

EXPN Vérification de l'existence d'une mailing list

HELP Demande d'informations

NOOP Vérification d'un dialogue avec le MTA

QUIT Demande de fermeture de la connexion

Syntaxe des réponses

Code numérique [chaîne de caractères]

500	Erreur de syntaxe (mot de commande inconnu)
501	Erreur de syntaxe des arguments
502	Commande non implémenté
503	Mauvaise séquence de commande
504	Paramètre non implémenté
211	Statut du MTA
214	Message d'aide
220	Nom de domaine du MTA + informations (en réponse à l'ouverture de la connexion)
221	Nom de domaine du MTA (fermeture de la liaison)
250	Requête acceptée
252	Impossibilité de vérifier une adresse (en réponse à un VRFY)
450	Pas de possibilité de délivrer le message (boîte de messagerie pleine)
550	Pas de possibilité de délivrer le message (adresse inexistante)
354	Message envoyé

Déroulement d'une Connexion

Etapes de connexion

Ouverture par le MUA d'une connexion TCP avec le MTA

Envoi par le MTA d'un message de bienvenue

220 nice to meet you <CRLF>

MUA: EHLO + identification du client <CRLF>

Envoi par le MTA de ses caractéristiques

MTA: 250-je m'appelle <CRLF>

250-caractéristique 1 <CRLF>

...

250 caractéristique n n <CRLF>

Echange de requêtes/réponses entre le MUA et le MTA

MUA: QUIT <CRLF>

MTA: 221 bla bla bla <CRLF>

Fermeture de la connexion par le MUA

Exemples de dialogue

Envoi d'un courrier

MUA: MAIL FROM:<@émetteur> + options <CRLF>

MTA: 250 OK <CRLF>

MUA: RCPT TO:<@destinataire> + options <CRLF>

MTA: 250 OK <CRLF>

MUA: DATA <CRLF>

MTA: 354 <CRLF>

MUA: message sur plusieurs lignes

MTA: . <CRLF>

MUA: 250 OK <CRLF>

Vérification d'une adresse

MUA: VRFY nom <CRLF>

MTA: 250 prénom nom <@électronique> <CRLF>

Ou alors, si ambiguïté

MTA: 553 <CRLF>

553- possibilité 1 <@1> <CRLF>

553- possibilité 2 <@2> <CRLF>

Méthode envoie

```
public void envoie(String in, String out, String sujet, String message)
{
    String str;
    try {
        str = from.readLine(); System.err.println(str);
        to.println("EHLO " + InetAddress.getLocalHost().getHostName());
        str = from.readLine(); System.err.println(str);
        to.println("MAIL FROM:<"+in+">");
        str = from.readLine(); System.err.println(str);
        to.println("RCPT TO:<"+out+">");
        str = from.readLine(); System.err.println(str);
        to.println("DATA");
        str = from.readLine(); System.err.println(str);
        to.println("Subject: " + sujet + "\n" + message + "\n.");
        str = from.readLine(); System.err.println(str);

        }catch( Exception e) {System.out.println("Erreur réseau");}
    }
```

1.2 PROGRAMMATION RESEAU

Test d'envoi du message

```
EnvoiMessage c = new EnvoiMessage("mailhost", 25);
String sujet = "test d'envoi d'un message";
String message = "cours du 4 mai 2016";
String expéditeur = "Claude.Montacie@paris-sorbonne.fr";
String destinataire = "ilgiil@laposte.net";
c.envoi(expéditeur, destinataire, sujet , message); c.fin();
```

Connexion établie entre /172.20.76.2:25 et mailhost/195.220.107.3:25

220 sorbon.sorbonne.fr ESMTP Sendmail 8.13.6/jtpda-5.4 ready at Tue, 4 May 2016 13:19:28 +0200 (CEST)

EHLO irp2 250-sorbon.sorbonne.fr Hello so-recherche.nat.univ-paris4.fr [195.220.213.3], pleased to meet you

MAIL FROM:<Claude.Montacie@paris4.sorbonne.fr> 250-ENHANCEDSTATUSCODES

RCPT TO:<ilgiil@laposte.net> 250-PIPELINING

DATA 250-8BITMIME

Subject: test d'envoi d'un message

cours du 25 avril 2006

. 250-SIZE 16777216

Description

Programme intégré dans le MUA

Machine locale ou site internet (web mail)

Deux services : transfert et gestion (tri, destruction)

Deux protocoles

Transfert puis gestion (pop)

MUA sur machine locale

Gestion puis transfert (imap)

MUA sur site internet

998 caractères au plus terminées par CR et LF

Mise en forme du message

Extraction des champs d'entête (subject, expéditeur, ..)

Trancodage mime pour les messages non texte (fichiers attachés)

Principes

Post Office Protocol version 3

1^{ère} version du protocole défini en 1984 (J. K. Reynolds) dans la RFC 918

Ouverture d'une connexion TCP bi-directionnelle sur le port 110

Suite de dialogues réseaux entre le MUA et le MDA

Dialogues (échange de requêtes / réponses)

Requête (par le MUA uniquement)

3/4 lettres + arguments

Réponse (par le MDA uniquement)

+OK -ERR + arguments / message

Trois états de la session

Authentification

Transaction

Mise à jour

Syntaxe des requêtes

Mots de commande

USER nom

PASS mot de passe

STAT Information sur les messages du MDA

LIST [n°] Affichage de la liste des messages

RETR n° Lecture d'un message donné

DELE n° Destruction différée (lors de la mise à jour) d'un message donné

RSET

NOOP Test de la connexion

QUIT Mise à jour des messages et fermeture de la connexion TCP

TOP n°X Affichage X lignes d'un message donné

UIDL [n°] Affichage du numéro d'identification de chaque message

APOP Méthode d'identification sécurisée

Exemple de dialogue

Passage dans l'état Transaction

MUA: USER mrose <CRLF>

MDA: +OK mrose is a real hoopy frood <CRLF>

MUA: PASS fsecret <CRLF>

MDA: -ERR invalid password <CRLF> // Mot de passe erroné

MUA: PASS tsecret <CRLF>

MDA: +OK mrose's maildrop has 2 messages (320 octets) <CRLF>

MUA: USER frated <CRLF>

MDA: -ERR sorry, no mailbox for frated here ... <CRLF> // Erreur d'identification

Passage dans l'état Mise à jour

MUA: DELE 1 <CRLF>

MDA: +OK message 1 deleted <CRLF>

MUA: QUIT <CRLF>

MDA: +OK dewey POP3 server signing off (maildrop empty) <CRLF>

2.2 PROGRAMMATION RESEAU

Méthode transaction

```
/** passage dans l'état transaction d'un utilisateur donné
 * @param user nom de l'utilisateur
 * @param password mot de passe de l'utilisateur
 */
void transaction (String user, String password) {
String str;
try {
    str = from.readLine(); System.err.println(str);
    to.println("USER " + user);
    str = from.readLine(); System.err.println(str);
    to.println("PASS " + password);
    str = from.readLine(); System.err.println(str);
    System.err.println("Passage dans l'état transaction");
} catch (Exception e) { System.err.println(e);}
}
```

2.2 PROGRAMMATION RESEAU

Méthode nbMessages

```
/** nombre de messages d'un utilisateur donné
 * @param user nom de l'utilisateur
 * @param password mot de passe de l'utilisateur
 * @return nombre de messages
 */
int nbMessages(String user, String password) {
String str; int nMessages = -1;
transaction(user, password);
    try {
        to.println("LIST");
        str = from.readLine(); System.err.println(str);
        do {
            str = from.readLine(); System.err.println(str);
            nMessages++;
        } while (!str.equals("."));
    } catch (Exception e) {System.err.println(e);}

    return nMessages++;
}
```

2.2 PROGRAMMATION RESEAU

Test de la méthode nbMessages

```
LectureMessage p = new LectureMessage("pop.laposte.net", 110);  
System.out.println(p.nbMessages("ilgii1", "milgii1"));  
p.fin();
```

```
Connexion établie entre /172.20.76.2:3743 et  
pop.laposte.net/81.255.54.8:110
```

```
+OK POP3 server ready (7.2.060.1)  
<9F5C76AAE346DE6624B181AF1A7CDA24FB08EAA8@mx.laposte.net>
```

```
USER ilgii1 +OK Password required
```

```
PASS milgii1 +OK 1 messages
```

```
Passage dans l'état transaction
```

```
LIST +OK
```

```
1 848
```

```
.
```

```
1
```

2.2 PROGRAMMATION RESEAU

Méthode lire

```
/** lecture d'un message donné d'un utilisateur donné
 * @param im index du message
 * @param user nom de l'utilisateur
 * @param password mot de passe de l'utilisateur
 */
String lire(int im, String user, String password) {
    String str = null, message = null;
    transaction(user, password);
    try {
        to.println("RETR " + im);
        str = from.readLine(); System.err.println(str);
        do { message += from.readLine() + "\n"; }
        while (!message.endsWith("\n.\n"));
    } catch (Exception e) { System.err.println(e);}

    return message;
}
```

2.2 PROGRAMMATION RESEAU

Test de la méthode lire (1/2)

```
LectureMessage p = new LectureMessage("pop.laposte.net", 110);  
System.out.print(p.lire(1, "ilgii1", "milgii1"));  
p.fin();
```

```
RETR 1 +OK 1007 bytes
```

```
bytesReturn-Path: <Claude.Montacie@paris4.sorbonne.fr>
```

```
Received: from smtp.laposte.net (10.150.9.34) by mx.laposte.net  
(7.2.060.1)
```

```
id 442BB2C500DF6421 for ilgii1@laposte.net; Wed, 3 May 2016  
10:35:52 +0200
```

```
X-IMELIOS: Suspected Spam
```

```
Received: from uranus.math-info.univ-paris5.fr (193.48.200.14) by  
smtp.laposte.net (7.3.105.2)
```

```
id 443A5D35007C9FB8 for ilgii1@laposte.net; Wed, 3 May 2016  
10:35:51 +0200
```

2.2 PROGRAMMATION RESEAU

Test de la méthode lire (2/2)

```
Received: from taittainger94 (taittainger94.math-info.univ-paris5.fr
[193.48.200.247])
        by uranus.math-info.univ-paris5.fr (8.13.6/jtpda-5.4) with
        ESMTP id k3L8ZpBK003499
        for <ilgiil@laposte.net>; Wed, 4 May 2016 10:35:51 +0200
Date: Wed, 4 May 2016 10:35:51 +0200
From: Claude.Montacie@paris4.sorbonne.fr
Message-Id: <200604210835.k3L8ZpBK003499@uranus.math-info.univ-
paris5.fr>
Subject: test d'envoi d'un message
X-Miltered: at uranus.math-info.univ-paris5.fr with ID 44489967.000 by
Joe's j-chkmail (http://j-chkmail.ensmp.fr)!

cours du 3 May 2016
.
```