Les protocoles SMTP et POP

I Présentation

Les protocoles SMTP (Simple Mail Transfert Protocol) et POP (Post Office Protocol) sont des protocoles les plus fondamentaux de l'Internet ; nous les utilisons tous les jours et nous sommes loin d'en connaître toutes les finesses.

Le protocole SMTP, qui se situe sur la couche "Application" du modèle OSI, est utilisé pour envoyer des courriers électroniques sur Internet.

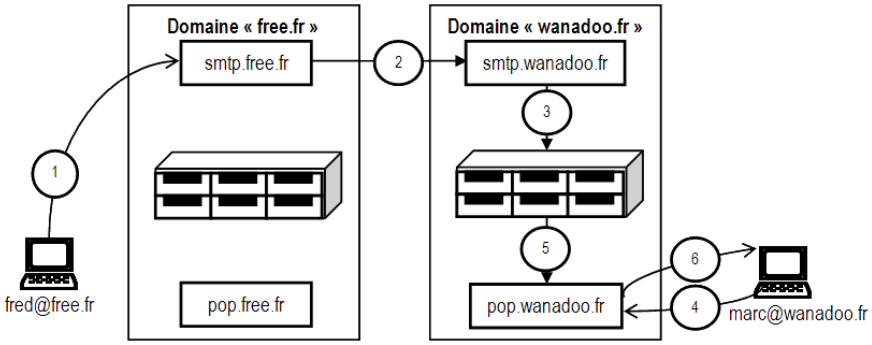
Le protocole POP définit le dialogue avec un service de lecture de boîtes aux lettres. D'autres protocoles fournissant ce service existent ; les principaux sont IMAP et DMSP. Le plus simple est le POP3 (version 3). Il est surnommé protocole "bureau de poste" car il est bien adapté aux petits volumes de courriers.

II Itinéraire d'un courrier électronique

II.1 Exemple

Supposons un cas simple mais cependant courant.

- ✗ Soit un abonné chez Free ayant une adresse électronique fred@free.fr. Fred dispose donc de deux serveurs :
  - ✔ smtp.free.fr pour envoyer des courriers,
  - ✔ pop.free.fr pour consulter ses courriers.
- ✗ Soit un autre abonné chez Wanadoo ayant une adresse électronique marc@wanadoo.fr. Marc dispose donc deux serveurs :
  - ✔ smtp.wanadoo.fr pour envoyer des courriers,
  - ✔ pop.wanadoo.fr pour consulter ses courriers.



Fred envoie un courrier électronique à Marc :

1. Fred compose le message avec son outil de messagerie préféré, disons Mozilla Thunderbird.
   Une fois que le message est composé, Fred clique sur le bouton "envoyer". Comme il a correctement configuré son outil, le message est envoyé sur le serveur smtp.free.fr.
2. Le serveur smtp.free.fr reçoit le message, constate que le destinataire n'est pas dans son domaine. Il cherche alors un serveur de messagerie dans le domaine wanadoo.fr et le trouve. Il envoie le message à smtp.wanadoo.fr
3. Le serveur smtp.wanadoo.fr reçoit le message, constate que le domaine est bien son domaine. Il range alors le message dans la boîte aux lettres de Marc. Il y restera aussi longtemps qu'il le faudra, sans le dire à personne.
4. Un jour, Marc décide de regarder s'il n'a pas de messages. Il envoie la requête à son serveur pop.wanadoo.fr.
5. Le serveur pop consulte alors la boîte aux lettres de Marc, constate qu'il y a un message dedans.
6. Il envoie à l'outil de messagerie de Marc qui, par défaut, demande à pop.wanadoo.fr de le supprimer de la boîte aux lettres. Mais c'est un comportement par défaut ; il est possible de demander à ne pas effacer les messages.

II.2 Définitions

Le **MUA** (Mail User Agent), c'est le client de messagerie (Outlook Express, Eudora, Pegasus, Mozilla Thunderbird, etc…).

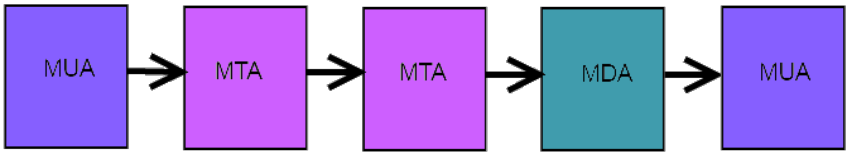
Le **MTA** (Mail Transfert Agent) est à prendre au sens plus large que le serveur SMTP. Le courrier peut être acheminé d'un point à un autre par l'intermédiaire d'agents de transfert qui ne gèrent pas de boîte aux lettres, mais savent relayer le courrier d'un point à un autre pour atteindre le serveur supportant les boîtes aux lettres. En effet, l'exemple vu en haut est le plus simple que l'on puisse imaginer. Dans la pratique, le courrier peut transiter par plusieurs MTA.

Le **MDA** (Mail Delivery Agent) est le service de remise du courrier dans les boîtes aux lettres des destinataires. Le MTA transmet au MDA les messages destinés aux clients du domaine.

Lorsque l'on rédige un courrier et qu'on le poste, on le fait avec le MUA qui le transmet au MTA qu'on lui a signalé dans la configuration (pour ceux qui sont abonné chez wanadoo, c'est normalement smtp.wanadoo.fr). C'est l'étape 1 du schéma.

De MTA en MTA, le message voyage jusqu'à celui qui a en charge la messagerie du domaine du destinataire (c'est l'étape 2). Il le passe alors (avec tous les autres messages entrant pour ce domaine) au MDA qui distribue ce courrier entrant dans les boîtes aux lettres concernées (étapes 3).

Les étapes 4, 5 et 6 concernent le serveur *POP* (ou *IMAP*).



II.3 Contenu d'un e-mail

Lorsque vous recevez un e-mail, votre MUA vous montre l'expéditeur, l'objet et le texte du message.

Mais votre e-mail contient toute une partie cachée qui nous permet de savoir quel chemin cet e-mail a suivi pour arriver à votre boîte aux lettres.

L'entête contient les informations pour :

- ✗ Identifier l'auteur du message ;
- ✗ Identifier le destinataire ;
- ✗ Savoir à qui il faut répondre ;
- ✗ Retrouver le chemin suivi par le message ;
- ✗ Savoir comment a été codé le message.

Le contenu du mail n'est pas visible par défaut mais il suffit de le sauvegarder et de le visualiser avec, par exemple, Wordpad.

Exemple :

X-Account-Key: account2
X-UIDL: 305-1336136797
X-Mozilla-Status: 0001
X-Mozilla-Status2: 00000000
X-Mozilla-Keys:
Return-path: <bmaheux@free.fr>
Received: from phuly.ac-creteil.fr (phuly.ac-creteil.fr [195.98.247.74])
by saigon.ac-creteil.fr
(Oracle Communications Messaging Exchange Server 7u4-18.01 64bit (built Jul 15
2010)) with ESMTP id <0M6V00BZOQZ08ME0@saigon.ac-creteil.fr> for
bmaheux@ac-creteil.fr; Mon, 09 Jul 2012 08:21:00 +0200 (CEST)
...
Received: from [127.0.0.1] (unknown [82.237.223.96])
by smtp1-g21.free.fr (Postfix) with ESMTP id B02A49400D3 for
<bmaheux@ac-creteil.fr>; Mon, 09 Jul 2012 08:20:56 +0200 (CEST)
Date: Mon, 09 Jul 2012 08:20:53 +0200
From: Barbara Maheux <bmaheux@free.fr>

Subject: Essai d'envoi de mail
To: Barbara Maheux <bmaheux@ac-creteil.fr>
Message-id: <4FFA7845.5080504@free.fr>
MIME-version: 1.0
Content-type: multipart/mixed; boundary="Boundary_(ID_AyzxjeX/XETbN20a5lD68Q) "
X-Greylist: whitelisted by SQLgrey-1.6.7
X-Antivirus: avast! (VPS 120708-1, 08/07/2012), Outbound message
X-Antivirus-Status: Clean
X-Virus-Scanned: amavisd-new at ac-creteil.fr
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:13.0) Gecko/20120614 Thunderbird/13.0.1
Original-recipient: rfc822;bmaheux@ac-creteil.fr
X-Spam-Status: No, hits=-0.307 tagged_above=-999 required=6.31 tests=AWL
X-Spam-Level:
X-Antivirus: avast! (VPS 120708-1, 08/07/2012), Inbound message
X-Antivirus-Status: Clean

This is a multi-part message in MIME format.

--Boundary_(ID_AyzxjeX/XETbN20a5lD68Q)
Content-type: text/plain; charset=ISO-8859-1; format=flowed
Content-transfer-encoding: 8bit
Ce mail ne sert qu'à voir le contenu d'un mail.

--Boundary_(ID_AyzxjeX/XETbN20a5lD68Q)
Content-type: text/x-vcard; charset=utf-8; name=bmaheux.vcf
Content-transfer-encoding: 7bit
Content-disposition: attachment; filename=bmaheux.vcf

begin:vcard
fn:Barbara Maheux
n:Maheux;Barbara
version:2.1
end:vcard

--Boundary_(ID_AyzxjeX/XETbN20a5lD68Q)--

Signification des champs de l'entête :

- ✗ **Received** : Chaque MTA qui reçoit le message y inscrit le nom du MTA qui lui a envoyé, ainsi que le sien. Il est ainsi possible de tracer complètement la route qu'a suivi le message de l'expéditeur au destinataire.
- ✗ **Message-ID** : C'est un identifiant unique du message. Il est attribué par le premier MTA qui reçoit le message (Protocole ESMTP : Extended SMTP).
- ✗ **From** : C'est l'adresse de l'expéditeur. Il est par défaut recopié dans le "Return-Path".
- ✗ **To :** C'est l'adresse du destinataire.
- ✗ **Subject** : L'objet du message.
- ✗ **Date** : La date d'émission écrite par le MUA de l'émetteur.
- ✗ **MIME-version :** Version du mode de codage des données.
- ✗ **Content-Type** : Type de codage utilisé.
- ✗ **Charset** : Jeu de caractères utilisé.
- ✗ **X-** : Tous les champs commençant par X- ne sont pas des champs officiels. Chaque MUA est libre d'ajouter autant qu'il veut. Leur contenu n'est pas pris en compte par les MTA.

III Protocole SMTP (RFC 2821)

Ce protocole permet d'acheminer (routage) et de délivrer le courrier électronique sur l'Internet. Il ne fait que transporter un message jusqu'à la boîte aux lettres, mais ne va pas plus loin.

Pour y arriver, il analyse dans un premier temps, la partie l'adresse située à droite du @ (à prononcer dans ce cas "at") pour trouver le domaine du destinataire. Si ce domaine le concerne, il cherche alors la boîte aux lettres du destinataire en regardant l'adresse située à gauche du @. Si le domaine ne le concerne pas, il va chercher le serveur SMTP qui gère ce domaine. Le courrier électronique est donc envoyé au travers du réseau grâce au protocole SMTP en passant par un ou plusieurs MTA et délivré à la machine de destination (serveur SMTP) par le même protocole. Le serveur final de courrier détermine si le message et livrable ou non et le dépose dans la boîte aux lettres du destinataire.

```
Telnet smtp.wanadoo.fr
220 mwinf2317.orange.fr ESMTP ABO ***************************
EHLO skyrunner.home
250-mwinf2317.orange.fr
250-PIPELINING
250-SIZE 10485760
250-AUTH PLAIN LOGIN
250-AUTH=PLAIN LOGIN
250 8BITMIME
MAIL FROM:<benja.roux@wanadoo.fr>
250 Ok
RCPT TO:<skyrunner63@hotmail.com>
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Date: Fri, 29 Dec 06 01:46:00 GMT
From: "Roux Benjamin" <benja.roux@wanadoo.fr>
To: "Skyrunner" <skyrunner63@hotmail.com>
Subject: Test SMTP

Test SMTP
.
250 Ok: queued as 86A867000081
```

III.1 Authentification

Les serveurs SMTP utilisent une authentification (login) avec un encodage particulier pour l'identification afin de ne pas transmettre les informations de login en clair sur le réseau et d'éviter les piratages.

Plusieurs types d'authentification peuvent être reconnue(s) par le serveur : PLAIN, LOGIN et CRAM-MD5. La réponse à la commande EHLO comprend le type ou les types d'authentification acceptées par le serveur.

Authentification PLAIN

Une méthode commune pour se connecter au serveur ESMTP est la méthode PLAIN. Il faut exécuter la commande "AUTH PLAIN" puis répondre le login avec mot de passe codé.

Exemple :

S: 220 smtp.server.com Simple Mail Transfer Service Ready
C: **EHLO localhost**
S: 250-smtp.server.com Hello client.example.com
S: 250-SIZE 1000000
S: 250 AUTH LOGIN PLAIN CRAM-MD5
C: **AUTH PLAIN**
S: 334
C: **dGVzdAB0ZXN0ADEyMzQ=**
S: 235 2.7.0 Authentication successful

Le login avec mot de passe codé est "\0login\0motdepasse" crypté en BASE64 (le \0 représente la valeur 0). Ce login peut être mis directement à la suite de la commande.

Authentification LOGIN

Cette méthode est une autre méthode qui nécessite de donner d'abord le login codé puis le mot de passe codé. Le codage utilisé est toujours le BASE64.

Exemple :

S: 220 smtp.server.com Simple Mail Transfer Service Ready
C: **EHLO localhost**
S: 250-smtp.server.com Hello client.example.com
S: 250-SIZE 1000000
S: 250 AUTH LOGIN PLAIN CRAM-MD5
C: **AUTH LOGIN**
S: 334 VXNlcm5hbWU6
C: **adlxdkej**
S: 334 UGFzc3dvcmQ6
C: **lkujsefxlj**
S: 235 2.7.0 Authentication successful

Authentification CRAM-MD5

L'inconvénient des deux premières méthodes est qu'il est facile de décoder le code BASE64 dont d'obtenir le login et le mot de passe.

Pour avoir une plus haute securitén le mécanisme d'authentification CRAM-MD5 peut être utilisé. CRAM-MD5 combine la réponse du serveur et le login et mot de passe afin d'obtenir un texte plus protégé. Ce code mélange du MD5 et du BASE64.

Exemple :

S: 220 smtp.server.com Simple Mail Transfer Service Ready
C: **EHLO localhost**
S: 250-smtp.server.com Hello client.example.com
S: 250-SIZE 1000000
S: 250 AUTH LOGIN PLAIN CRAM-MD5
C: **AUTH CRAM-MD5**
S: 334 PDQxOTI5NDIzNDEuMTI4Mjg0NzJAc291cmNlZm91ci5hbmRyZXcuY211LmVkdT4=
C: **cmpzMyBlYzNhNTlmZWQzOTVhYmExZWM2MzY3YzRmNGI0MWFjMA==**
S: 235 2.7.0 Authentication successful

III.2 Commandes de base

Voici les commandes de bases :

- ✗ **HELO nom_de_domaine** : permet au client de s'identifier (EHLO pour les serveurs de type ESMPT)
- ✗ **MAIL FROM:adr_mail_expéditeur** : indique l'expéditeur
- ✗ **RCPT TO:adr_mail_destinataire** : indique le destinataire
- ✗ **DATA** : les lignes suivantes forment le corps du message

La fin du message est signalée par une ligne contenant juste un point

- ✗ **QUIT** : termine la session et ferme la connexion
- ✗ **RSET** : annule la transaction en cours
- ✗ **VRFY nom** : vérifie qu'un utilisateur est géré par le serveur SMTP
- ✗ **EXPN list** : affiche les membres d'une liste de fusion
- ✗ **HELP [command]** : affiche des informations sur les commandes disponibles ou une commande particulière

Le but du protocole de courrier électronique Simple Mail Transfert Protocol SMTP est de transférer du courrier électronique selon un procédé efficace et fiable. Le client SMTP établit une communication bidirectionnelle vers un serveur SMTP ; celui-ci peut être soit la destination finale, soit seulement un intermédiaire. Il s'agit d'un protocole fonctionnant en mode connecté encapsulé dans une trame TCP/IP sur le port 25.

Dès que le canal de transmission est établi, un échange de procédure s'assure que le serveur SMTP est bien à l'écoute du client.

Le protocole SMTP fonctionne grâce à des commandes textuelles envoyées au serveur SMTP. Chacune des commandes envoyées par le client, validé par la chaîne de caractères ASCII "CR/LF", est suivie d'une réponse de serveur SMTP composée d'un numéro et d'un message descriptif.

Voici un scénario de demande d'envoi de mail à un serveur SMTP :

- ✗ Lors de l'ouverture de la session SMTP, la première commande à envoyer est la commande HELO ou EHLO suivie d'un espace et du nom de domaine du serveur *SMTP* :
  EHLO nom_de_domaine
- ✗ La seconde commande est la MAIL FROM suivie de l'adresse email de l'expéditeur. Si cette commande est acceptée, le serveur envoie le message "250 OK" :
  MAIL FROM: adresse_mail_expéditeur
- ✗ La commande suivante est la RCPT (Recipient) suivie de l'adresse mail du destinataire (plusieurs commandes RCPT à suivre si plusieurs destinataires du message). Si la commande est acceptée, le serveur envoie le message "250 OK" :
  RCPT TO: adresse_mail_destinataire
- ✗ La commande DATA est la troisième étape de l'envoi. Elle annonce le début du corps du message. Si la commande est acceptée le serveur envoie un message numéroté 354 indiquant que l'envoi du corps de mail peut commencer et considère que l'ensemble des lignes suivantes jusqu'à la fin du message repérée par une ligne contenant uniquement par un point.

Le corps du mail contient éventuellement certaines des entêtes suivantes :

- ✗ **Date** : date du message (possibilité d'antidater vos mails...). Si aucune date n'est spécifiée, le premier MTA s'en charge.
- ✗ **From** : indique l'expéditeur.
- ✗ **To** : indique les destinataires.
- ✗ **Cc** : indique les destinataires en copie.
- ✗ **Bcc** : indique les destinataires en copie cachée.
- ✗ **Subject** : sujet du message

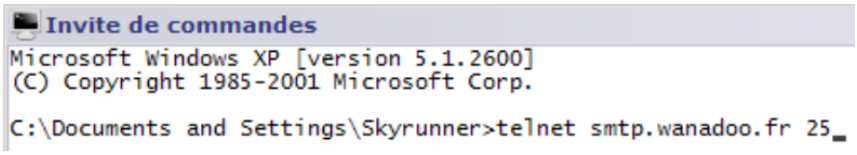
Si la commande est acceptée, le serveur envoie le message "250 OK".

Attention : il faut mettre obligatoirement une ligne blanche après la partie Subject.

III.3 Envoi de mail grâce à telnet

Il est possible d'envoyer un courrier grâce à telnet sur le port 25 du serveur SMTP.

Exemple de connexion :



Exemple d'envoi de mail :

IV Protocole POP

Le protocole POP (Post Office Protocol) que l'on peut traduire par "Protocole de bureau de poste" permet, comme son nom l'indique, d'aller récupérer son courrier sur un serveur POP distant. Il existe deux principales versions de ce protocole, POP2 et POP3, auxquels sont affectés respectivement les ports 109 et 110. Ils fonctionnent à l'aide des commandes textuelles radicalement différentes.

Tout comme dans le cas du protocole SMTP, le protocole POP (POP2 et POP3) fonctionne grâce à des commandes textuelles envoyées au serveur POP. Chacune des commandes envoyées par le client, validée par la séquence CR/LF, est composée d'un mot clé éventuellement accompagné d'un ou plusieurs arguments et est suivie d'une réponse du serveur POP composée d'un numéro et d'un message descriptif.

Voici les commandes de base :

- ✗ **Connexion :**
  - ✔ **USER toto@titi.fr** : permet la connexion de l'utilisateur toto
  - ✔ **PASS pwd** : indique le mot de passe de l'utilisateur

  Si la connexion est possible, le serveur indique OK.
- ✗ **Exploitation :**
  - ✔ **STAT** : retourne le nombre de messages et le nombre d'octets de l'ensemble de ces messages
  - ✔ **LIST** : retourne la liste des messages (le numéro et la taille de chacun)
  - ✔ **RETR n** : récupération du message numéro n
  - ✔ **DELE n** : marquage du message n pour effacement
  - ✔ **RSET** : retire le marquage "effacement" aux messages marqués par DELE
  - ✔ **QUIT** : effacement des messages marqués et fermeture de la connexion
  - ✔ **NOOP** : ne rien faire, utile pour ne pas perdre la connexion et éviter un « délai d'attente dépassé »
- ✗ **Options :**
  - ✔ **TOP n x** : affiche x lignes du message numéro n
  - ✔ **UIDL** : affiche (pour un seul ou pour tous les messages) un identifiant unique qui ne varie pas entre chaque session.

Le protocole POP3 gère aussi l'authentification à l'aide d'un nom utilisateur et d'un mot de passe. Il n'est par contre pas sécurisé car les mots de passe, au même titre que les mails, circule en clair sur le réseau. D'autre part, le protocole POP3 bloque la boîte aux lettres lors de la consultation. Ce qui signifie qu'une consultation simultanée par deux utilisateurs d'une même boîte aux lettres est impossible.

Exemple :

```
+OK <18074.961492882@pop3.free.fr>
USER ******************          Normalement, le texte apparaît en clair...
+OK


PASS *******                     Ici également!
+OK


STAT
+OK 46 132860                    46 messages, 132 Ko


LIST
+OK
1 753                            La liste éa été tronquée...
2 3023                           Il y a au total 46 lignes
3 2444
   .
   .
   .
45 2659
46 2487                          Notez le point final sur les réponses
.                                de plus d'une ligne.


TOP 2 1
+OK 3023 octets
Return-Path: <ccwmrs-admin@listes.3base.com>
Delivered-To: online.fr-christian.caleca@free.fr
Received: (qmail 6392 invoked from network); 15 Jun 2000 11:47:08 -0000
Received: from ns.3base.com (64.29.16.216)
  by mrelay2.free.fr with SMTP; 15 Jun 2000 11:47:08 -0000
Received: from ns (localhost [127.0.0.1])
        by ns.3base.com (8.9.3/8.9.3) with ESMTP id HAA25316;
        Thu, 15 Jun 2000 07:46:36 -0400
Received: from mouarf.dhs.org (ca-ol-marseille-4-224.abo.wanadoo.fr [62.161.99.22
4])
```