

Réseaux – TD 1

Découverte de quelques commandes et exploration de protocoles

Janvier 2007

Exercice 1 Expérimentations sur les services réseau

1. Déterminez le port associé au service *discard*.
2. En utilisant successivement les commandes `host`, `nslookup` et `dig`, déterminez l'adresse IP de `www.free.fr`, puis de `www.univ-paris7.fr`. Quels sont les noms d'hôtes associés aux adresses obtenues ?
3. À l'aide de la commande `dig`, déterminer les serveurs de courrier électronique du réseau `ufr-info-p7.jussieu.fr`. (voir la page de manuel de `dig`).
4. À l'aide de la commande `telnet`, déterminez l'heure qu'il est sur la machine `lenteja`.
5. À l'aide de la commande `finger`, affichez les informations de quelques utilisateurs. Ensuite interrogez le service `finger` à l'aide de `telnet`.

Exercice 2 SMTP

Le protocole SMTP (*Simple Mail Transfer Protocol*, RFC 821, port 25) sert à envoyer du courrier électronique (*e-mail*) à des utilisateurs locaux ou distants. Il s'agit d'un protocole dit *request-response*, dans lequel le client envoie une commande au serveur puis attend la réponse de ce dernier.

Remarque : les techniques utilisées dans cette partie permettent, en théorie, d'envoyer des mails en se faisant passer pour quelqu'un d'autre. Une telle activité serait non seulement discourtoise, mais aussi illégale, et je vous déconseille fortement de sous-estimer les capacités d'investigation de nos administrateurs système.

Les commandes SMTP les plus utiles sont les suivantes :

- « `HELO machine` » : à envoyer au début d'une connexion SMTP. La chaîne *machine* est le nom du hôte à partir duquel vous vous connectez.
- « `MAIL FROM: utilisateur` » : commence une transaction SMTP visant à envoyer un message. La chaîne *utilisateur* est l'adresse de l'auteur du message (de la forme *user@domain*).
- « `RCPT TO: utilisateur` » : déclare un destinataire de la transaction courante ; peut être répétée plusieurs fois.
- « `DATA` » : déclare le début du message, qui commence sur la ligne suivante et se termine par un point « `.` » tout seul sur une ligne. Cette commande a pour effet de terminer la transaction courante.
- « `QUIT` » : termine une connexion SMTP.

Le message passé à la commande « `DATA` » doit avoir le format défini par le RFC 822. Il doit commencer par une série d'entêtes (« `From:` », « `To:` », « `Subject:` », etc.) suivis d'une ligne vide suivie du corps du message lui-même.

1. En vous connectant directement au port SMTP d'une machine à l'aide de la commande « `telnet` », envoyez un mail à l'un de vos collègues.
2. Répétez l'expérience précédente en donnant des adresses différentes dans l'enveloppe (commande « `RCPT TO:` » de SMTP) et dans le message (entête « `To:` » de RFC 822). Que se passe-t-il ?

Exercice 3 Clients Java

En Java, il est possible d'ouvrir une connexion en utilisant la classe `Socket`. Par exemple :

```
Socket s = new Socket(InetAddress.getByName("smtp.free.fr"), 25);
```

Ensuite les méthodes `getOutputStream` et `getInputStream` permettent de créer des flux (de types `OutputStream` et `InputStream`) pour écrire et lire sur cette connexion.

1. Écrivez un programme Java qui se connecte au service `smtp` et envoie un email.
2. Écrivez un programme Java qui se connecte au service `daytime` d'une machine courante et affiche l'heure donnée par celle-ci.
3. Écrivez un programme Java qui prend en argument un nom de hôte (par exemple `www.ufr-info-p7.jussieu.fr`) ainsi qu'une « partie locale » d'une URL (par exemple `/index.html`) puis envoie à cet hôte une requête HTTP de la forme

```
GET /index.html HTTP/1.0
Host: www.ufr-info-p7.jussieu.fr
```

où chaque ligne est terminée par la suite `"\r\n"`. Il écrira ensuite la réponse obtenue sur la sortie standard.

4. Répétez l'exercice suivant en filtrant les entêtes de la réponse du serveur.