
TD1, TD2 Notion de protocole - les cas ftp et pop3

Exercice 1 : Le plus simple : en utilisant un navigateur

La plupart des navigateurs modernes supportent le protocole FTP. Dans l'exercice, le serveur FTP sera `ftp.ens-lyon.fr`.

1. Commencez par vous connecter à ce serveur avec un navigateur web classique (Netscape, Mozilla, konqueror, Internet Explorer, Opera, ...) en donnant l'adresse suivante et promenez vous dans l'arborescence en cliquant sur les répertoires.

```
ftp ://ftp.ens-lyon.fr/
```

2. Rendez vous dans le répertoire FTP `/pub/LIP/Rapports/` et ouvrez le fichier `Publications.LIP`. Lorsque vous cliquez sur le nom du fichier, votre navigateur envoie une série de requêtes au serveur afin que celui-ci envoie le contenu du fichier demandé. Lorsqu'il est téléchargé, le contenu du fichier est affiché sur votre écran.

Exercice 2 : Utilisation classique en ligne de commande

Maintenant, explorons l'interface en ligne de commande.

1. A partir de votre fenêtre telnet, lancez la commande

```
ftp ftp.ens-lyon.fr
```

Vous devez à présent vous identifier sur ce serveur. Vous n'avez pas de compte sur cette machine, vous ne pouvez donc pas taper votre login et mot de passe habituels. Cependant, FTP propose un mode anonyme afin d'autoriser les personnes ne possédant pas de compte à accéder à une partie restreinte du disque de la machine (la partie publique du serveur FTP).

Connectez-vous sous le nom `anonymous`. Donnez votre adresse E-mail comme mot de passe, quelque chose comme `dupond@emi.u-bordeaux.fr`. (en fait n'importe quelle chaîne de caractères qui contient un `@` convient)

2. Une fois que vous êtes connecté, demandez la liste des commandes disponibles avec la commande `help`.
3. Changez de répertoire au moyen de la commande `cd` afin de vous placer dans le répertoire

```
/pub/LIP/Rapports/
```

4. Demandez la liste des fichiers disponibles avec la commande `ls -l`. Le serveur vous répond :

```
227 Entering Passive Mode (140,77,168,4,12,48).
150 Opening ASCII mode data connection for file list
[...]
-rw-r--r--  1 0          root          1148440 Apr  7  2000 Publications.LIP
[...]
226-Transfer complete.
226 Quotas off
```

5. Demandez par exemple à recevoir le fichier `Publications.LIP` au moyen de la commande `get`. Tapez `get Publications.LIP`. Le serveur vous répond quelque chose qui ressemble à :

```
local: Publications.LIP remote: Publications.LIP
227 Entering Passive Mode (140,77,168,4,12,51).
150 Opening BINARY mode data connection for Publications.LIP (1148440
bytes).
100% |*****| 1121 KB 257.82 KB/s
00:00 ETA
226 Transfer complete.
1148440 bytes received in 00:04 (252.12 KB/s)
```

- Sortez de l'interface par la commande `quit`, et vérifiez dans votre répertoire que ce fichier a bien été créé et regardez son contenu (`more Publications.LIP`).

Exercice 3 : RFC 959 : File Transfer Protocol (FTP) - Spécification

La RFC 959 est disponible en ligne à :

<http://www.w3.org/Protocols/rfc959>

- expliquez brièvement le mode de fonctionnement d'un transfert de fichier par FTP.
- établir la liste des commandes `ftp` en précisant leur équivalent `unix`, lorsqu'il existe.
Exemple : `MKD (MaKe Directory) <-> mkdir`
- donner les différents mode de transmission, en précisant leurs avantages et inconvénients.
- établir la liste des classes principales de code erreur. Comment une application client peut-elle exploiter ces informations ?

Exercice 4 : Client `gftp`

Pendant toutes les manipulations ci-dessous, observer les commandes `ftp` effectuées par `gftp` (fenêtre du bas ou journal de bord).

- utiliser le client `gftp` pour télécharger le fichier `Publications.LIP` du premier exercice
- télécharger tous les fichiers de `/pub/LIP/Rapports` et les placer dans un de vos répertoire. Le téléchargement des fichiers est simultané : comment le protocole permet-il cela ?
- modifier un des fichiers téléchargés. Vérifier que `Outils/Comparer les fenêtres` vous permet de détecter la modification. Ainsi `gftp` permet de synchroniser facilement deux répertoires distants. `gftp` compare-t-il les contenus ou simplement les dates de création des fichiers et leurs tailles ?

Exercice 5 : Configuration du démon `ftp`

La page

<http://www.liafa.jussieu.fr/~yunes/internet/ftp/>

fait une excellente synthèse de ce que vous venez de voir. Elle contient de plus des informations sur le paramétrage d'un démon `ftp`...

Exercice 6 : Protocole POP3

`Telnet` est un protocole simple de connexion à distance : il permet de transmettre des caractères entre une machine locale (écran+clavier) et une machine distante.

- Utiliser un client `telnet` pour vous connecter sur une machine du département. Par défaut le port utilisé est 23. Il est possible de préciser un numéro de port autre que 23 : `telnet M N`. Dans ce cas le client `telnet` se connecte au processus de la machine `M` associé au numéro de port `N`.
`POP3` est un protocole pour la consultation d'une boîte aux lettres distante : il est exploité par les lecteurs de courriers (`Outlook Express`, `Eudora`, le lecteur de `Netscape`...) et utilise le port 110.
- Lisez votre courrier en utilisant un client `pop3`.
- Connectez vous au service `pop3` de `free` en essayant ceci : `telnet pop.free.fr 110`. Il faut ensuite donner une ligne de commande, terminée par `CR/LF` (en pratique, un retour chariot suffit).
Voici quelques commandes du protocole `POP3` :

| Commande | Fonction |
|----------|---|
| USER | Nom du compte de l'utilisateur |
| PASS | mot de passe |
| STAT | Donne le nombre de messages non lus et leur taille totale |
| RETR n | Récupère le message numéro n |
| DELE n | Supprime le message numéro n |
| LAST | Donne le numéro du dernier message auquel on a accédé |
| LIST [n] | Taille de n^{me} message ou de tous les messages |
| RSET | Annule la suppression de tous les messages |
| TOP n k | Affiche les entêtes et k lignes du messages numéro n |
| NOOP | Ne fait rien |
| QUIT | Termine la session POP3 |

- Lisez votre courrier sur free en connectant un client telnet au service pop3, en tant qu'utilisateur mstinfo1 (mot de passe : master). Vous pouvez vous inspirer du scénario suivant qui, après une phase d'authentification, demande le transfert d'un message :


```
S>+OK QPOP (version 2.53) at boite.info.iuta.u-bordeaux.fr starting.
C>USER felix
S>+OK Password required for felix.
C>PASS xxxxxxxxx
S>+OK felix has 2 messages (2828 octets).
C>STAT
S>+OK 2 2828
C>LIST
S>+OK 2 messages (2828 octets)
S>1 1823
S>2 1005
S>.
C>RETR 2
S>+OK 1005 octets
S>Return-Path: <felix@labri.fr>
S>Received: from smtp16.wanadoo.fr (smtp16.wanadoo.fr [193.252.19.183])
S>Status: RO
S>
S>coucou
S>pf
S>.
```

Remarques sur POP3 :

Le client POP3 envoie des lignes de commande (ou requêtes) et le serveur répond par une ligne commençant par +OK ou -ERR. Tout dialogue commence par une phase d'authentification (nom et mot de passe). Après une phase d'authentification réussie, le client peut consulter sa boîte aux lettres en émettant des requêtes permettant les opérations suivantes :

- Demande du nombre de messages,
- Demande de la taille de chaque message,
- Lecture d'un message

Toute réponse d'un serveur POP3 qui contient plusieurs lignes est terminée par une ligne contenant un caractère '.' tout seul en début de ligne. Si, 'par malchance', une ligne de données commence par un point, celui-ci est doublé. Ainsi, on ne pourra pas confondre une ligne de données qui ne contiendrait qu'un point (elle sera alors codée "..CR/LF") avec la ligne de fin (".CR/LF"). Pour plus d'informations, voir la RFC !

Exercice 7 : Protocole POP3 : connexion à un serveur à l'aide de sockets

Il s'agit maintenant d'écrire un tout petit client `clientPOP` en java capable de récupérer les messages d'un serveur POP3.

Exercice 8 : Dialogue avec un serveur ftp avec telnet

Un serveur FTP (File Transfer Protocol) est un serveur qui permet aux utilisateurs qui se connectent à lui de

télécharger des fichiers. Dans l'exercice, le serveur FTP sera `ftp.lip6.fr`.

Le but de cet exercice est de prendre la place de votre navigateur pour envoyer vous-mêmes les requêtes au serveur. La séquence de requêtes qui suit est probablement très proche de celle que fait votre navigateur de manière transparente lorsque vous traversez les répertoires du serveur et téléchargez des fichiers.

1. Dans votre terminal, tapez `telnet ftp.lip6.fr 21` pour vous connecter au port 21, qui est le numéro de port du service FTP. Attendez que le serveur soit ready.

```
> telnet ftp.lip6.fr 21
Connected to nephtys.lip6.fr.
Escape character is '^]'.
220 ftp.lip6.fr FTP server ready.
```
2. Vous devez à présent vous identifier sur ce serveur. Comme vous n'avez pas de compte personnel, identifiez vous en mode anonyme par les requêtes `USER` (nom d'utilisateur, ici `ftp`) et `PASS` (mot de passe, ici votre adresse e-mail).

```
USER ftp
331 Guest login ok, send your complete e-mail address as password.
PASS xxxx@emi.u-bordeaux.fr
```
3. Vous êtes connectés au serveur. La requête `CWD` (*Change Working Directory*) vous permet de changer de répertoire (comme `cd`). `PWD` (*Print Working Directory*) fonctionne... comme d'habitude.
4. La requête `LIST` provoque l'envoi de la liste des fichiers du répertoire courant (comme `ls -l`). Si vous envoyez la requête `LIST`, celle-ci échoue. En effet, le serveur FTP envoie ses données sur une connexion différente, et aucune n'a été mise en place pour le moment.
5. Ouvrez une seconde fenêtre `telnet` pour avoir un nouvel interpréteur de commandes. C'est ici que vous allez ouvrir la seconde connexion au serveur.
6. Depuis la première fenêtre, passez en mode passif en envoyant la requête `PASV`. Le serveur vous renvoie une série de nombres séparés par des virgules. Les quatre premiers correspondent à l'adresse IP du serveur FTP (vous pouvez le vérifier dans votre second terminal avec `dig`). Les deux derniers permettent de connaître un numéro de port, celui sur lequel il faut vous connecter pour recevoir les données. Soient a et b les deux derniers nombres. Le numéro de port est $256a + b$ (utilisez une calculatrice). Exemple : si $a = 240$ et $b = 106$, le port est $240 \times 256 + 106 = 61546$.
7. Depuis le second terminal, lancez `telnet` en lui donnant en paramètre l'adresse IP correspondante aux quatre premiers nombres et comme numéro de port, celui que vous venez de calculer.
Exemple : si le serveur a renvoyé les nombres `140,77,1,24,240,106`, lancez `telnet 140.77.1.24 61546`.
8. Votre connexion est prête, vous pouvez envoyer la requête `LIST` depuis votre premier terminal. Le contenu du répertoire s'affiche dans le second terminal et la connexion est fermée.
9. Pour recevoir un fichier, envoyez de nouveau une requête `PASV` au serveur qui vous indique six nouveaux nombres. Calculez le nouveau port et relancez le `telnet` de la seconde fenêtre. Depuis la première fenêtre, envoyez la requête `RETR nom_de_fichier`.

Exercice 9 : client FTP en java Il s'agit maintenant d'écrire un tout petit client `clientFTP` en java capable de se connecter à un serveur et de télécharger un fichier. Votre classe devra :

- accepter en paramètre un nom de serveur et un nom de fichier à télécharger ;
- se connecter au port 21 du serveur spécifié ;
- basculer en mode passif et établir une seconde connexion pour récupérer le contenu du fichier en le sauvegardant en local ;
- afficher les réponses du serveur `ftp` dans une fenêtre dédiée