

# Chapitre IV Partie 2

## FTP

*Cours Réseaux Avancés M1 SIC-IA*

Ilyas Bambrik

# Table des matières



<b>I - FTP (File Transfer Protocol)</b>	3
<b>II - Fonctionnement FTP</b>	4
<b>III - Liste des commandes FTP et réponses serveurs</b>	6
<b>IV - TFTP (Trivial FTP)</b>	7
<b>Références</b>	8

# FTP (File Transfer Protocol)



FTP est une solution de partage et transfert de fichiers ( par exemple : archivage, stockage distribué, etc). Comme TELNET, le développement de ce protocole a fait face aux différents représentations de fichiers et des hétérogénéités des systèmes d'exploitation.

FTP est bâti sur les fonctionnalités suivantes :

- *Authentification des utilisateurs* : Les utilisateurs doivent être authentifiés et selon leurs identités, les opérations permises pour ces derniers seront définies ( par exemple : certains utilisateurs auront les droits d'écriture alors que d'autres seront limités à la lecture);
- *Mode interactif / automatique* : FTP été conçu pour permettre aux utilisateurs un accès interactif via l'ensemble de commandes FTP, ainsi qu'un accès automatique aux applications.
- *Spécification du format du transfert* ;

# Fonctionnement FTP



- FTP utilise TCP (*port 21 et 20*) au niveau de la couche transport et repose sur une architecture client / serveur ;
- Comme d'autres protocoles applicatifs (HTTP par exemple), FTP crée un processus esclave pour traiter une nouvelle connexion ;
- Contrairement à la plus part des protocoles applicatifs, ce protocole utilise deux connexions TCP :
  1. *Une connexion permettant la transmission des commandes de contrôles* : La première connexion TCP (*plus précisément une connexion TELNET*) établie entre le client et le serveur est appelée *connexion de contrôle*. Cette connexion est faite d'un port quelconque (dynamiquement alloué comme d'habitude) de la part du client vers un *port 21 (qui est le port réservé pour FTP)*. Cette connexion primaire est maintenue jusqu'à la fin de la session FTP et permet de transmettre des commandes FTP depuis le client vers le serveur. De l'autre coté, le serveur interprète les commandes FTP et transmet un code de réponse vers le client ;
  2. *Une connexion de transmission de données* : La deuxième connexion permet de transmettre les données ( par exemple lors de la demande d'obtention d'un fichier par client). Cette connexion par contre est temporaire et sera clôturée après la fin de la transmission des données. Typiquement, le port utilisé pour la transmission des données du *coté serveur est le port 20 qui est réservé pour FTP* ;
- Deux modes de transmission de données sont possibles :
  1. *Mode actif* : Dans ce mode, le client transmet le numéro de port, par lequel il souhaite recevoir les données / fichier, vers le serveur par la connexion de contrôle. Par la suite, le serveur établie lui même la connexion TCP avec le client via ce port (ce qui sort un peut de la tradition des protocoles applicatifs). Par la suite, la transmission du fichier commence immédiatement ;
  2. *Mode Passif* : Dans le mode passif le client envoie une commande *PASV* sur la connexion de contrôle pour demander au serveur le numéro de port qui doit être utilisé pour la transmission de données. Par la suite, le client connecte au port spécifié par le serveur et ce dernier commence directement la transmission du fichier demandé;
- L'utilisateur peut spécifier que le transfert du fichier doit se faire en mode binaire (*copie identique octet par octet comme les images et fichiers .exe*) ou mode ASCII pour que le système d'exploitation du transmetteur converti la représentation vers une représentation normalisée avant de transmettre le contenu.

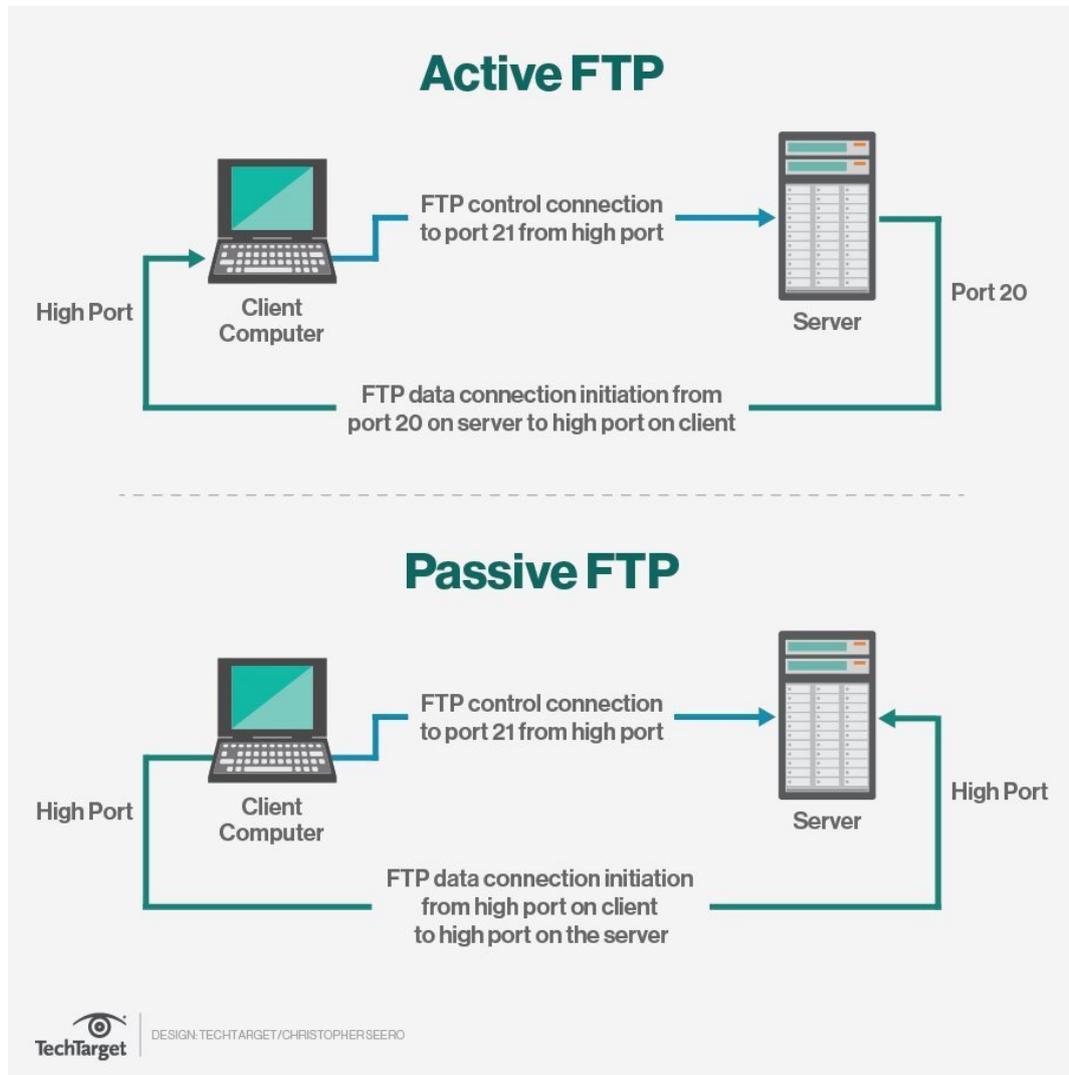


Figure IX.1. Modes FTP<sup>9\*</sup>

**⚠ Attention : Mode ASCII et mode binaire**

Attention, l'utilisation du mode ASCII pour transmettre un fichier binaire (image par exemple) donne un fichier inutilisable du côté du récepteur. Car la conversion de la représentation interne vers la représentation standard change le contenu du fichier.

**📦 Complément : Mode actif et TELNET**

- Le mode actif nécessite que le serveur établie la connexion vers le client (c.à.d le serveur envoie un segment TCP avec SYN allumé vers le client). Les connexions entrantes sont souvent bloquées par le par-feu car ils sont considérés comme une tentative d'attaque;
- FTP utilise TELNET pour standardiser la représentation des commandes (NVT) mais n'utilise pas la négociation d'options TELNET. Ce-ci permet de transmettre les commandes dans un format que le serveur peut interpréter ;

# Liste des commandes FTP et réponses serveurs



Les commandes FTP sont regroupés dans *trois catégories* :

- Commandes d'accès (*USER, PASS, CWD, QUIT, CDUP, REIN*);
- Commandes de spécification des paramètres (*PORT, PASV, TYPE, MODE*);
- Commandes du service FTP (*RETR, STOR, ALLO, REST, DELE, MKD, ABOR, NOOP, STAT*);

De l'autre part, selon le déroulement de la commande, *le serveur répond avec un code de trois chiffres et un message explicatif*. En outre, chaque chiffre dans la réponse du serveur a une signification :

- Le premier chiffre indique le succès ou l'échec (permanent ou temporaire) de la commande. Ce premier chiffre peut prendre de 1 à 5 ;
- Le deuxième chiffre divise les catégories de réponses définies par le premier chiffre en cinq sous catégories pour donner plus de précision sur la réponse. La valeur de ce chiffre indique la réponse concerne quoi ( exemple *si le deuxième chiffre est égale à 3, ceci indique que la réponse est relative à l'authentification, si la valeur est égale à 2, la réponse est relative à la syntaxe de la commande* ) ;
- Le dernier chiffre donne dix codes pour chaque sous-catégorie ;

# TFTP (Travail FTP)

## IV

- TFTP est un protocole de transmission de fichiers simplifié qui ne prend compte ni de l'authentification, ni de l'accès interactif ;
- Le client initie la transmission de fichier par un simple message contenant le nom du fichier et le type d'opération (*OpCode* == 1 pour dire que le client souhaite recevoir le fichier, *OpCode* == 2 pour indiquer que le client souhaite transmettre un fichier vers le serveur) ;
- La valeur du champ *Mode* indique le mode de transmission (ASCII ou binaire) ;
- TFTP transmet les morceaux du fichier en taille de 512 octets. Ainsi, la réception d'un morceau de données de taille inférieur à 512 signale automatiquement la fin du fichier. En outre, les messages contenant des parties du fichier transféré ont *OpCode* == 3 ;
- Contrairement à FTP, TFTP utilise UDP et assure l'acquittement des messages au niveau de la couche application. *OpCode* == 4 signale un message ACK pour le numéro du bloque indiqué dans le message. En utilisant UDP, TFTP peut diffuser le contenu dans le réseau locale ;
- *OpCode* == 5 signale une erreur de transmission ;

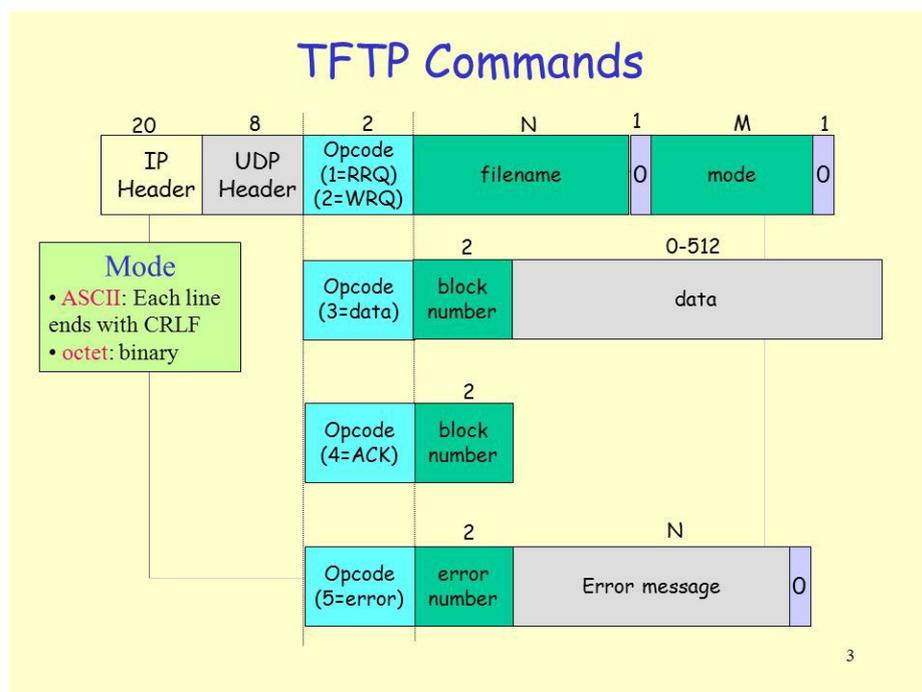


Figure IX.2. Protocole TFTP

