FTP

INTRODUCTION

Introduction

- Protocole FTP (File Transfert Protocol)
- Objectifs :
  - de promouvoir le partage de fichiers (programmes informatiques et/ou données),
  - d'encourager l'utilisation indirecte ou implicite (via des programmes) d'ordinateurs distants,
  - de prémunir l'utilisateur contre les variations de formats de stockage de données entre les différents hôtes,
  - de transférer les données d'une façon efficace et fiable.
- FTP, bien que directement utilisable par un utilisateur depuis un terminal, est néanmoins conçu essentiellement pour être utilisé par des programmes.

Historique

- La mise en place du protocole FTP date de 1971, date à laquelle un mécanisme de transfert de fichiers (décrit dans le RFC 141) entre les machines du MIT (Massachussetts Institute of Technology) avait été mis au point.
- De nombreux RFC ont ensuite apporté des améliorations au protocole de base, mais les plus grandes innovations datent de juillet 1973.
- Le protocole FTP est actuellement défini par le RFC 959 (File Transfer Protocol (FTP) - Specifications).

RFC

- Que signifie RFC ?
  - Les RFC (Request For Comments) sont un ensemble de documents qui font référence auprès de la Communauté Internet et qui décrivent, spécifient, aident à l'implémentation, standardisent et débattent de la majorité des normes, standards, technologies et protocoles liés à Internet et aux réseaux en général.
- Qui écrit et gère les RFC ?
  - La suite de protocoles TCP/IP représente un ensemble de normes établies par un organisme qui s'appelle l'IETF (Internet Engineering Tasking Force). Ceux-ci publient officiellement leurs rapports sous formes de requêtes, disponibles pour tous, permettant d'éclaircir un grand nombre de sujets relatifs à TCP/IP.

RFC

- Chacun de ces documents représente une proposition de spécification qui peut à tout moment être rendue obsolète par un nouveau document RFC. Ainsi, les RFC sont des fichiers textes dont le nom est "rfcxxxx.txt" dont xxxx est un nombre incrémenté pour chaque nouveau RFC. Il en existe actuellement plus de 2000.
- En réalité n'importe qui peut écrire une RFC et la soumettre à l'IETF en la transmettant au responsable: rfc.editor@rfc.editor.org. Si celle-ci est acceptée, elle paraîtra après avoir été vérifiée par les responsables. La RFC1543, intitulée instructions to RFC authors, explique comment rédiger une RFC.

Spécification	RFC
Protocole UDP	RFC768
Protocole IP	RFC791
Protocole ICMP	RFC792
Protocole TCP	RFC793
Protocole FTP	RFC959
Internet Mail	RFC822
Protocole Telnet	RFC854
Protocole NNTP	RFC977
Netbios	RFC1001
Protocole SLIP	RFC1055
MIB	RFC1156
TCP/IP	RFC1180

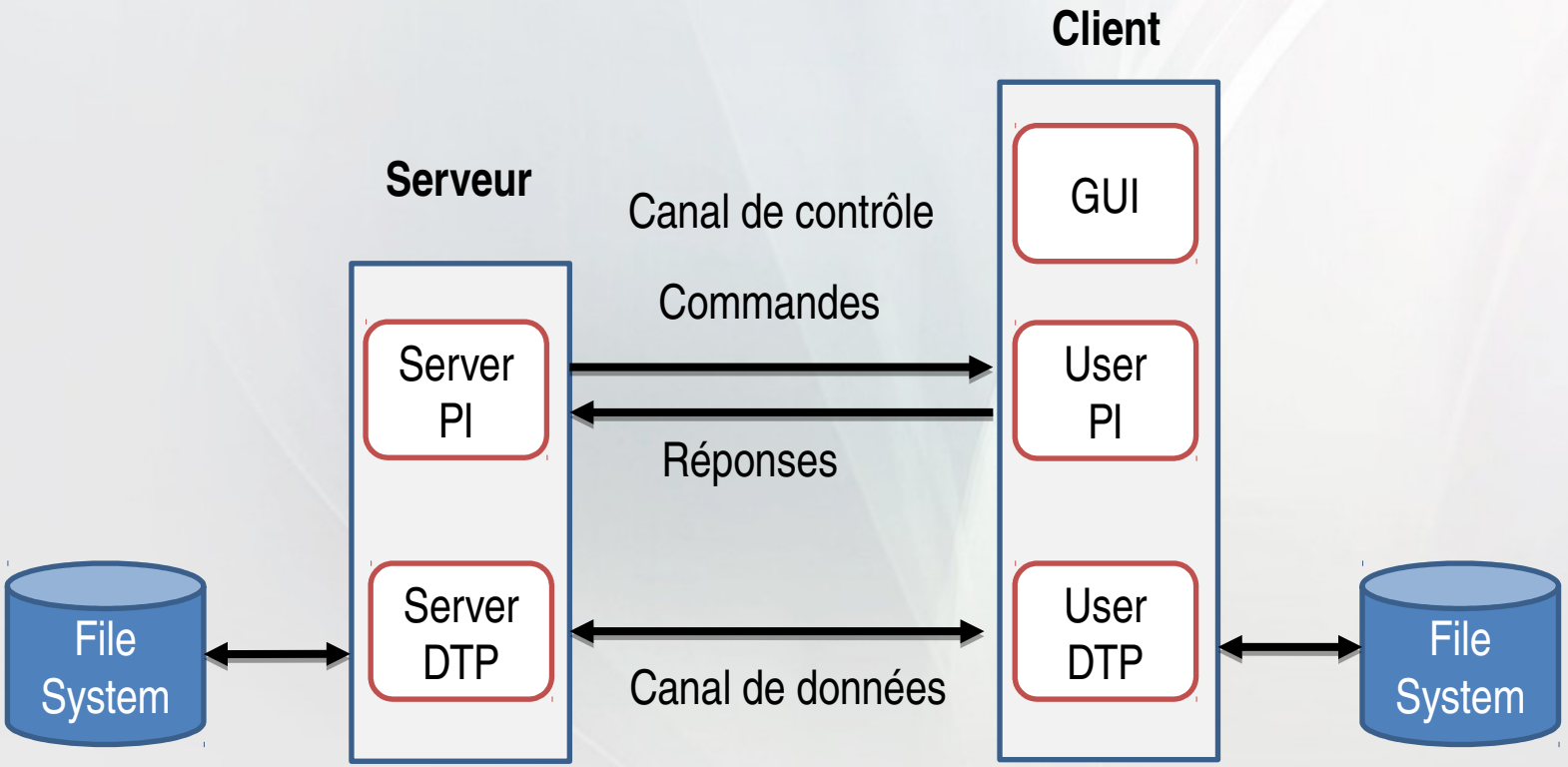
MODÈLE FTP

Modèle FTP

- Le protocole FTP s'inscrit dans le modèle client-serveur
- Lors d'une connexion FTP, deux canaux de transmissions sont ouverts :
  - un canal pour les commandes (canal de contrôle, port 21)
  - un canal pour les données

Modèle FTP

Schéma de principe :



Modèle FTP

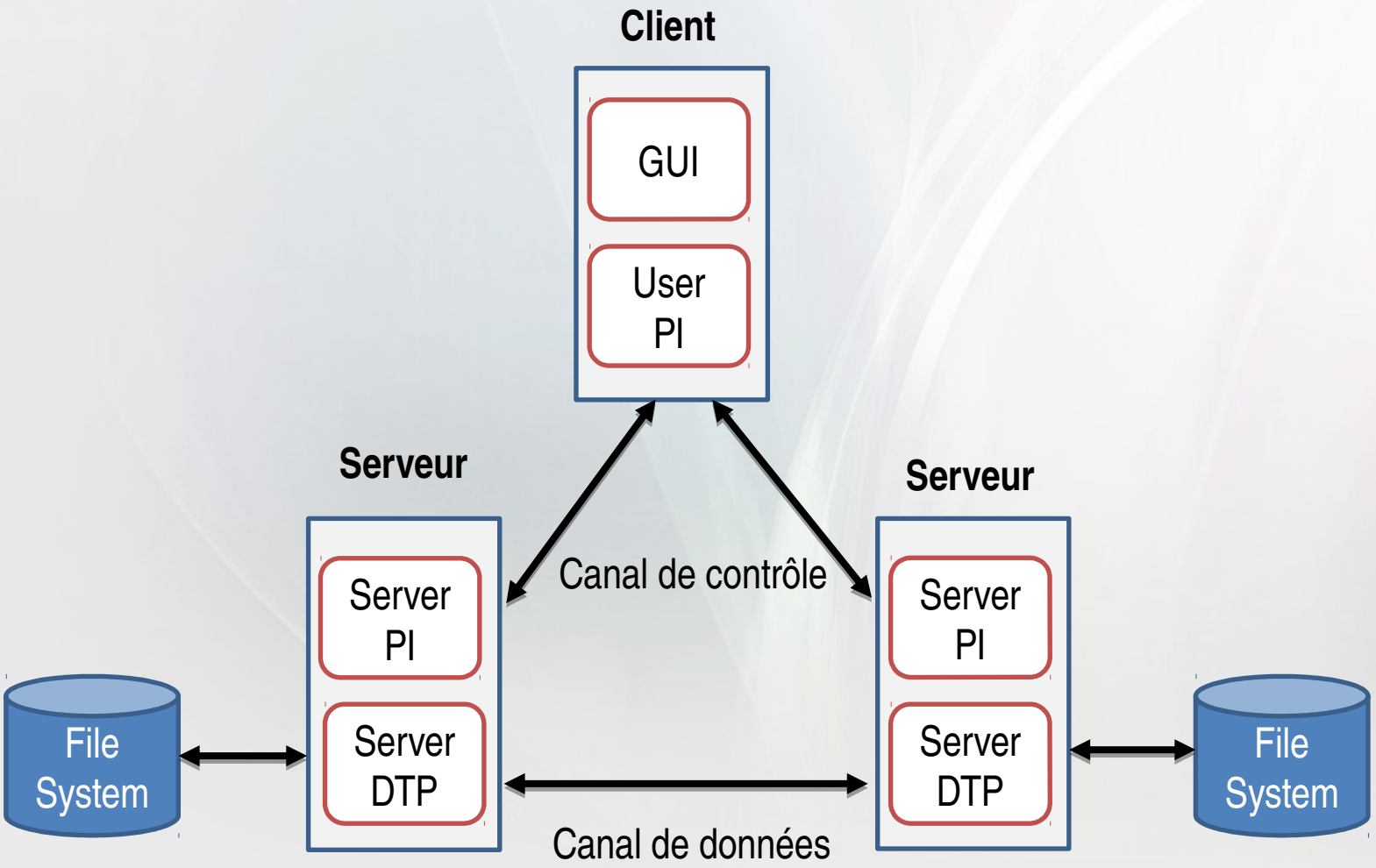
- Deux processus :
  - le DTP (Data Transfer Process) est le processus chargé d'établir la connexion et de gérer le canal de données. Le DTP côté serveur est appelé SERVER-DTP, le DTP côté client est appelé USER-DTP
  - Le PI (Protocol Interpreter) est l'interpréteur de protocole permettant de commander le DTP à l'aide des commandes reçues sur le canal de contrôle. Il est différent sur le client et sur le serveur :
    - Le SERVER-PI est chargé d'écouter les commandes provenant d'un USER-PI sur le canal de contrôle sur un port donné, d'établir la connexion pour le canal de contrôle, de recevoir sur celui-ci les commandes FTP de l'USER-PI, d'y répondre et de piloter le SERVER-DTP
    - Le USER-PI est chargé d'établir la connexion avec le serveur FTP, d'envoyer les commandes FTP, de recevoir les réponses du SERVER-PI et de contrôler le USER-DTP si besoin

Modèle FTP

- Lors de la connexion d'un client FTP à un serveur FTP, le USER-PI initie la connexion au serveur selon le protocole Telnet. Le client envoie des commandes FTP au serveur, ce dernier les interprète, pilote son DTP, puis renvoie une réponse standard. Lorsque la connexion est établie, le serveur-PI donne le port sur lequel les données seront envoyées au Client DTP. Le client DTP écoute alors sur le port spécifié les données en provenance du serveur.

- Il est important de remarquer que, les ports de contrôle et de données étant des canaux séparés, il est possible d'envoyer les commandes à partir d'une machine et de recevoir les données sur une autre. Ainsi, il est par exemple possible de transférer des données entre deux serveurs FTP en passant par un client pour envoyer les instructions de contrôle.

Modèle FTP


Client
GUI
User PI
Serveur
Server PI
Server DTP
Canal de contrôle
Serveur
Server PI
Server DTP
File System
File System
Canal de données

Modèle FTP

- Remarque :
  - Dans cette configuration, le protocole impose que les canaux de contrôle restent ouverts pendant tout le transfert de données. Ainsi un serveur peut arrêter une transmission si le canal de contrôle est coupé lors de la transmission.

LES COMMANDES FTP

les commandes

- Toutes les communications effectuées sur le canal de contrôle suivent les recommandations du protocole Telnet. Ainsi les commandes FTP sont des chaînes de caractères Telnet (en code NVT-ASCII) terminées par le code de fin de ligne Telnet (c'est-à-dire la séquence <CR>+<LF>, Carriage Return (retour chariot) suivi du caractère Line Feed, notée <CRLF>).

- Si la commande FTP admet un paramètre, celui-ci est séparé de la commande par un espace (<SP>).

les commandes

- Les commandes permettent de définir :
  - Le port utilisé;
  - Le mode de transfert des données;
  - La structure des données;
  - La nature de l'action à effectuer.

- On distingue trois types de commandes :
  - Les commandes de contrôle d'accès;
  - Les commandes de paramétrage de transfert;
  - Les commandes de services FTP.

les commandes

- Commande de contrôle d'accès :
  - USER : Chaîne de caractères permettant d'identifier l'utilisateur. L'identification de l'utilisateur est nécessaire pour établir une communication sur le canal de données.
  - PASS : Chaîne de caractères spécifiant le mot de passe de l'utilisateur. Cette commande doit être immédiatement précédée de la commande *USER*. Il revient au client de masquer l'affichage de cette commande pour des raisons de sécurité.
  - ACCT : Chaîne de caractères représentant le compte (account) de l'utilisateur. Cette commande n'est généralement pas nécessaire. Lors de la réponse à l'acceptation du mot de passe, si la réponse est 230 cette phase n'est pas nécessaire, si la réponse est 332, elle l'est.

les commandes

- CWD : *Change Working Directory* : cette commande permet de changer le répertoire courant. Cette commande nécessite le chemin d'accès au répertoire à atteindre comme argument.
- CDUP : *Change to Parent Directory* : cette commande permet de remonter au répertoire parent. Elle a été introduite pour remédier aux problèmes de nommage de répertoire parent selon les système (généralement "..")
- SMNT : Cette commande permet de monter un volume sous un système de fichier différent sans changer de contexte pour la session. Les paramètres de transfert sont de même inchangés. L'argument est un chemin d'accès valide du système local.

les commandes

- REIN : Cette commande tue une connexion USER, libérant toute les ressources d'entrées/sorties et les informations de session, sauf pour l'opération de transfert en cours qui est achevée normalement. Tous les paramètres sont rétablis dans leurs valeurs par défaut et le canal de contrôle est laissé ouvert. L'état obtenu est identique à l'état dans lequel serait un canal de contrôle juste après son établissement. Une commande USER est en général attendue.
- QUIT : Commande permettant de terminer la session en cours. Le serveur attend de finir le transfert en cours le cas échéant, puis de fournir une réponse avant de fermer la connexion.

les commandes

- Commande de paramètres de transfert :
  - PORT : Chaîne de caractères permettant de préciser le numéro de port à utiliser.
  - PASV : Commande permettant d'indiquer au serveur DTP de se mettre en attente une connexion sur un port spécifique choisi aléatoirement parmi les ports disponibles. La réponse à cette commande est l'adresse IP de la machine et le port.
  - TYPE : Cette commande permet de préciser le type de format dans lequel les données seront envoyées.
  - STRU : Caractère Telnet précisant la structure du fichier (F pour File, R pour Record, P pour Page).
  - MODE : Caractère Telnet précisant le mode de transfert des données (S pour Stream, B pour Block, C pour Compressed).

les commandes

- Commande de service FTP :
  - RETR : Cette commande (*RETRIEVE*) demande au serveur DTP une copie du fichier dont le chemin d'accès est passé en paramètre.
  - STOR : Cette commande (*store*) demande au serveur DTP d'accepter les données envoyées sur le canal de données et de les stocker dans le fichier portant le nom passé en paramètre. Si le fichier n'existe pas, le serveur le crée, sinon il l'écrase
  - STOU : Cette commande est identique à la précédente, si ce n'est qu'elle demande au serveur de créer un fichier dont le nom est unique. Le nom du fichier est retourné dans la réponse
  - APPE : Grâce à cette commande (*append*) les données envoyées sont concaténées dans le fichier portant le nom passé en paramètre s'il existe déjà, dans le cas contraire il est créé

les commandes

- ALLO : Cette commande (*allocate*) demande au serveur de prévoir un espace de stockage suffisant pour contenir le fichier dont le nom est passé en argument.
- REST : Cette commande (restart) permet de reprendre un transfert là où il s'était arrêté. Pour cela cette commande envoie en paramètre le marqueur représentant la position dans le fichier à laquelle le transfert avait été interrompu. Cette commande doit être immédiatement suivie d'une commande de transfert.
- RNFR : Cette commande (*rename from*) permet de renommer un fichier. Elle indique en paramètre le nom du fichier à renommer et doit être immédiatement suivie de la commande *RNTO*
- RNTO : Cette commande (rename to) permet de renommer un fichier. Elle indique en paramètre le nom du fichier à renommer et doit être immédiatement précédée de la commande RNFR

les commandes

- ABOR : Cette commande (*abort*) indique au serveur DTP d'abandonner tous les transferts associés à la commande précédente. Si aucune connexion de données n'est ouverte, le serveur DTP ne fait rien, sinon il la ferme. Le canal de contrôle reste par contre ouvert.
- DELE : Cette commande (delete) permet de supprimer le fichier dont le nom est passé en paramètre. Cette commande est irrémédiable, seule une confirmation au niveau du client peut être faite.
- RMD : Cette commande (remove directory) permet de supprimer un répertoire. Elle indique en paramètre le nom du répertoire à supprimer.
- MKD : Cette commande (make directory) permet de créer un répertoire. Elle indique en paramètre le nom du répertoire à créer.
- PWD : Cette commande (print working directory) permet de renvoyer le chemin complet du répertoire courant.

les commandes

- LIST : Cette commande permet de renvoyer la liste des fichiers et répertoires présents dans le répertoire courant. Cette liste est envoyée sur le DTP passif. Il est possible de passer en paramètre de cette commande un nom de répertoire, le serveur DTP enverra la liste des fichiers dans le répertoire passé en paramètre.
- NLST : Cette commande (name liste) permet d'envoyer la liste des fichiers et répertoires dans le répertoire courant.
- SITE : Cette commande (site parameters) permet au serveur de proposer des services spécifiques, non définis dans le protocole FTP.
- SYST : Cette commande (*system*) permet d'envoyer des informations sur le serveur distant.

les commandes

- STAT : Cette commande (status) permet d'émettre l'état du serveur, par exemple pour connaître la progression d'un transfert en cours. Cette commande accepte en argument un chemin d'accès, elle retourne alors les mêmes informations que LIST mais sur le canal de contrôle.
- HELP : Cette commande permet de connaître l'ensemble des commandes comprises par le serveur. Les informations sont retournées sur le canal de contrôle.
- NOOP : Cette commande (no operations) sert uniquement à obtenir une commande OK du serveur. Elle peut servir uniquement pour ne pas être déconnecté après un temps d'inactivité trop élevé.

LES RÉPONSES FTP

Les réponses FTP

- Les réponses FTP permettent d'assurer la synchronisation entre client et serveur FTP. Ainsi à chaque commande envoyée par le client, le serveur effectuera éventuellement une action et renverra systématiquement une réponse.
- Les réponses sont constituées d'un code à 3 chiffres indiquant la façon suivant laquelle la commande envoyée par le client a été traitée. Toutefois, ce code à 3 chiffres étant difficilement lisible par un humain, il est accompagné d'un texte (chaîne de caractères Telnet séparée du code numérique par un espace).

Les réponses FTP

- Les codes de réponse sont constitués de 3 chiffres dont voici les significations :
  - Le premier chiffre indique le statut de la réponse (succès ou échec) ;
  - Le second chiffre indique ce à quoi la réponse fait référence ;
  - Le troisième chiffre donne une signification plus spécifique (relative à chaque deuxième chiffre).

Les réponses FTP

Premier chiffre		
Chiffre	**Signification**	**Description**
1yz	Réponse préliminaire positive	L'action demandée est en cours de réalisation, une seconde réponse doit être obtenue avant d'envoyer une deuxième commande
2yz	Réponse positive de réalisation	L'action demandée a été réalisée, une nouvelle commande peut être envoyée
3yz	Réponse intermédiaire positive	L'action demandée est temporairement suspendue. Des informations supplémentaires sont attendues de la part du client
4yz	Réponse négative de réalisation	L'action demandée n'a pas eu lieu car la commande n'a temporairement pas été acceptée. Le client est prié de réessayer ultérieurement
5yz	Réponse négative permanente	L'action demandée n'a pas eu lieu car la commande n'a pas été acceptée. Le client est prié de formuler une requête différente

source : www.commentcamarche.net

Les réponses FTP

Second chiffre		
Chiffre	**Signification**	**Description**
x**0**z	Syntaxe	L'action possède une erreur de syntaxe, ou bien il s'agit d'une commande non comprise par le serveur
x**1**z	Information	Il s'agit d'une réponse renvoyant des informations (par exemple pour une réponse à une commande STAT)
x**2**z	Connexions	La réponse concerne le canal de données
x**3**z	Authentification et comptes	La réponse concerne le login (USER/PASS) ou la demande de changement de compte (CPT)
x**4**z	Non utilisé par le protocole FTP	
x**5**z	Système de fichiers	La réponse concerne le système de fichiers distant

source : www.commentcamarche.net

MODE DE TRANSMISSION

Mode de transmission

- FTP définit trois modes : un qui formate les données est permet de recommencer la transmission si nécessaire; un qui compresse en plus les données pour un transfert plus efficace; et un dernier mode qui laisse passer les données avec le moins d'encodage possible.
- Tous les transferts de données doivent s'achever par la transmission d'une séquence de fin-de-fichier (EOF), la quelle peut être explicite, ou implicitement déduite de la fermeture du canal

Mode de transmission

- Mode Flux : Les données sont transmises comme un flux d'octets. Il n'y a dans ce cas aucune restriction sur la représentation des données utilisée.
- Mode Bloc : Le fichier est transmis comme une suite de blocs de données précédés d'un ou plusieurs octets d'en-tête. L'en-tête contient un champ de comptage de blocs, et un code de description. Le champ de comptage indique la longueur totale du bloc de données en octets, et indique donc le début du bloc suivant.
- Mode compressé

Info

- Pour plus d'information :
  - http://www.httr.ups-tlse.fr/pedagogie/annexes/tcp-ip/rfc959.html#ap3

Linux et FTP

- Serveur FTP :
  - **ProFTPd :**
    - licence GNU GPL
    - chroot
    - Bien documenté, config proche d'Apache
    - compatible IP6
    - version 1.3.5 (15 mai 2014)
    - supporte SLL/TLS (FTPS)
    - http://www.proftpd.org/

- **Pure-FTPd :**
  - Licence BSD
  - environnement chroot
  - contrôle bande passante
  - quotas
  - authentification LDAP,PAM,SQL
  - SSL/TSL
  - version 1.0.36 (mars 2012)
  - http://www.pureftpd.org/project/pure-ftpd

- **VsFTPd :**
  - Very Secure FTP Daemon : orienté sécurité
  - développé par Chris Evans qui gère la sécurité de Google Chrome
  - séparation des privilèges
  - configuration facile
  - bande passante
  - IP6
  - SSL
  - licence GNU
  - version 3.0.2 (septembre 2012)
  - les plus sécurisé d'où mon choix !!