SERVEURS FTP

AVERTISSEMENT :

La lecture de ce document nécessite la connaissance préalable du service FTP décrit dans le module « Services Internet ».

SOMMAIRE

.....1

Chapitre 1 : Présentation des Serveurs.....4

1.Introduction 4

2.Protocole FTP.....4
2.1Origine et références.....4
2.2Rôles du protocole.....4
2.3Modèle FTP.....4
2.4Commandes du protocole.....6
2.5Réponses du serveur.....8

3.Types de sites.....10
3.1Site privé.....11
3.2Site public.....11
3.3Organisation des sites publics.....11

4.Serveurs.....11

5.Configurer un serveur en trois étapes.....13
5.1Identité du serveur ou du site.....13
5.2Nombre d'utilisateurs connectés.....14
5.3Attribution des droits d'accès.....14

6.Compléments.....15
6.1Procédé FXP.....15
6.2Protocole Secure FTP.....15

7.Conclusion 15

Chapitre 2 : Serveurs Linux.....17

1.Serveurs sous UNIX.....17
1.1.Démon ftpd standard.....17
1.2.Présentation de Wu-Ftpd et ProFtpd.....19
1.3.Procédures d'installation.....19
1.4.Environnement de fonctionnement.....22

2.Serveur wu-ftpd.....26
2.1.Sites de référence.....26
2.2.Plate-formes et versions.....26
2.3.Arborescence.....26
2.4.Service public.....26
2.5.Service privé.....28
2.6.Administration.....31

3.Serveur Proftpd.....37
3.1.Sites de références.....37
3.2.Plate-formes et versions.....37
3.3.Arborescence et complément.....38
3.4.Paramètres globaux.....38
3.5.Site public.....40
3.6.Serveur Virtuel.....48
3.7.Administration.....50
3.8.Conclusion.....55

Chapitre 3 : Serveur Windows Serv-U.....57

1.Présentation.....57
1.1.Caractéristiques.....57
1.2.Editions 57

2.Installation.....58
2.1.Conseils préalables.....58

2.2.Installation proprement dite....58
2.3.Configuration initiale....60
2.4.Fichiers installés....64

3.Administration....66
3.1.Présentation....66
3.2.Licence....68
3.3.Identité du serveur....69
3.4.Paramètres globaux (Settings)....70
3.5.Activité du serveur....75
3.6.Domaine d'appartenance du serveur....76

Annexe 1 : Administration locale KDE de Wu-FTP....89

1.Classe d'utilisateurs....89

2.Dossiers....90

3.Sécurité....90

4.Messages....91

5.Journalisation....92

6.Ratios....92

7.Envois....93

8.Hôtes virtuels....93

Annexe 2 : Administration distante Webmin pour Wu-FTP....95

1.Menu général....95

2.Users and Classes....96

3.Message and Banners....96

4.Limits and Access Control....97

5.Networking....98

6.Logging....98

7.Aliases and Paths....99

8.Anonymous FTP....99

9.Permissions....100

10.Miscellaneous Options....100

11.Fichiers de Configuration du Logiciel....101

Annexe 3 : Administration distante Webmin de ProFTP....102

1.Menu général....102

2.Configuration globale....103
11.1Networking Options....103
11.2Logging Options....104
11.3Files and Directories....105
11.4Access Control....106
11.5Miscellaneous....106
11.6Authentification....107
11.7Per-Directory Options Files....108
11.8Denied FTP Users....109
11.9Edit Config Files....109

3.Serveurs virtuels....110
11.10Serveur par défaut....111
11.11Virtual Server....115

4.Fichiers de Configuration du Logiciel....119

Chapitre 1 : Présentation des Serveurs

1. Introduction

Le service FTP représente un des services **les plus anciens** du réseau Internet puisqu'il date des années 70.

Il est le premier outil mis à la disposition des utilisateurs pour **échanger des fichiers** sur des réseaux fonctionnant sur les **protocoles TCP/IP**. Il permet d'accéder à des milliers et des milliers de serveurs de par le monde, proposant un nombre incalculable de fichiers en libre service.

Comme tout service Internet de première génération, il s'inscrit dans un **modèle client-serveur**. En utilisant le service, l'utilisateur envoie depuis un logiciel client situé sur son ordinateur, des requêtes à un serveur. Ce dernier, situé sur un autre ordinateur, attend les demandes pour effectuer des actions, avant de déclencher le transfert.

Le serveur et le client communiquent par l'intermédiaire du **protocole FTP** (**File Transfert Protocol**). Comme son nom l'indique, il s'agit d'un protocole de transfert de fichiers dont le **numéro de port de service TCP est 21**.

2. Protocole FTP

2.1 Origine et références

La mise en place du protocole FTP date de 1971, date à laquelle un mécanisme de transfert de fichiers (décrit dans les RFC 114 et RFC 141) entre les machines du MIT (*Massaschussetts Institute of Technology*) avait été mis au point.
Mis en œuvre au sein du réseau ARPANET, ancêtre du réseau INTERNET, de nombreux RFC ont ensuite apporté des améliorations au protocole de base, mais les plus grandes innovations datent de juillet 1973. Le protocole a atteint son développement final en 1985.

Le protocole FTP est actuellement défini par le **RFC 959** (*File Transfer Protocol (FTP) - Specifications*) auquel il faut rajouter deux extensions sur la sécurité et l'internationalisation.

2.2 Rôles du protocole

Le protocole FTP définit la façon de transférer les données sur un réseau TCP/IP. Le protocole FTP a pour objectifs de permettre :

- un partage de fichiers entre machines distantes,
- une indépendance aux systèmes de fichiers des machines clientes et serveur,
- de transférer des données de manière efficace.

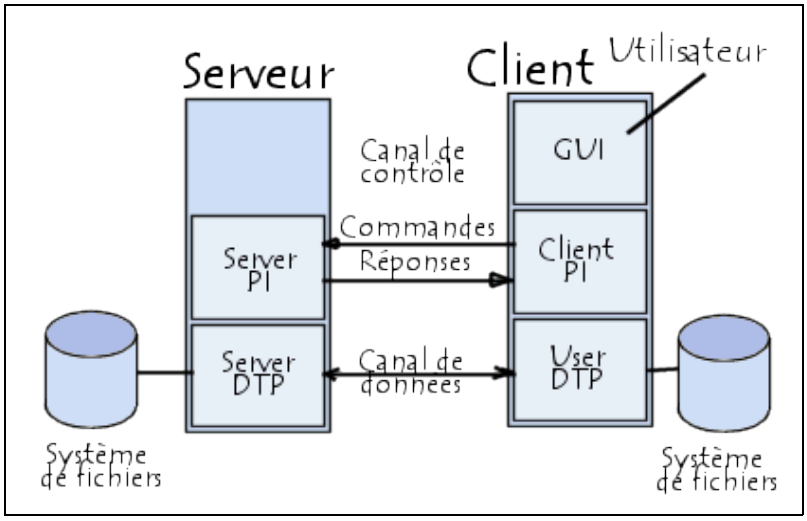
2.3 Modèle FTP

Lors d'une connexion FTP, deux canaux de transmission sont ouverts :

- Un canal pour les **commandes** (canal de contrôle) sur le **port 21**,

- Un canal pour les **données** sur le **port 20**.

Le **RFC 959** prévoit l'utilisation de ces numéros de port. Depuis l'explosion du réseau Internet, l'augmentation des débits constatée, les serveurs et clients emploient d'**autres numéros de port pour transférer les données**.
S'il avait fallu attendre que le premier client libère la connexion pour que le deuxième eusse pu transférer ses données, alors le service n'eut jamais eu autant de succès. Dans les années 70 et 80, cette attente n'était pas gênante. En revanche, les réseaux actuels garantissent connexions rapides et taux de transmission élevés, sans parler des réseaux Intranet où la bande passante est censée être entièrement disponible.
C'est la raison pour laquelle le **RFC 1123** relativise les affirmations du RFC 959 en ce qui concerne l'emploi des deux numéros conventionnels. Le **changement cyclique des adresses de port** permet d'exploiter au maximum la capacité de transmission des réseaux existants : par exemple, les ports situés **entre 2048 et 2148**. Une fois le dernier numéro de port de l'intervalle utilisé (2148), le programme reprend le premier numéro de port de l'intervalle.



Pour chaque canal de transmissions associé à un numéro de port, le client comme le serveur possède un processus permettant de gérer chacun des deux types d'informations :

- 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 en provenance 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.
- le **DTP** (*Data Transfer Process*) est le processus chargé d'établir la connexion pour gérer le canal de données. Le DTP côté serveur est appelé *SERVER-DTP*, le DTP côté client est appelé *USER-DTP*.

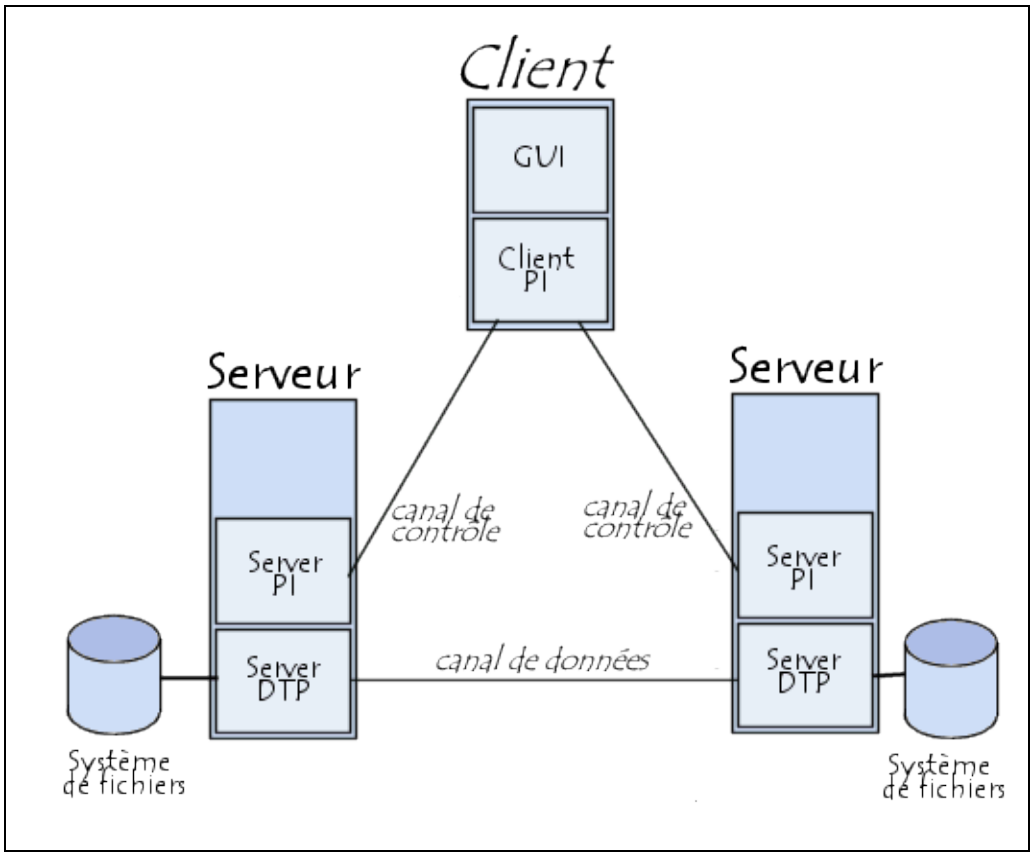
Lors de la connexion d'un client FTP à un serveur FTP, le USER-PI initie la connexion au serveur selon le **protocole Telnet**. Le client envoie des commandes FTP au serveur. Ce

dernier les interprète, pilote son DTP, puis renvoie une réponse standard. Lorsque la connexion est établie, le serveur-PI donne le numéro de 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 station donnée 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 et en transférant les informations entre deux processus serveurs connectés sur le bon port.

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 du transfert.

Cette possibilité a été exploitée par les logiciels utilisant le procédé FXP.



2.4 Commandes du protocole

Les commandes qui suivent sont les véritables commandes interprétées par les deux processus évoqués ci-avant.
Le client FTP, par l'intermédiaire du navigateur, du logiciel client spécialisé, ou par l'intermédiaire d'une console terminale n'utilise pas ces commandes.
Seul l'administrateur (ou le hacker) peut être amené à utiliser ces commandes, à travers deux consoles différentes, par le biais d'une connexion à distance *telnet*.

Ces commandes standards de la liaison de commande sont regroupées en différentes catégories :

- Les commandes de contrôle d'accès :

USER	identifie le nom d'utilisateur
PASS	identifie le mot de passe de l'utilisateur
ACCT	demande une identification supplémentaire (compte ou *account*)
CWD	modifie la position de l'utilisateur dans l'arborescence du serveur
CDUP	permet à l'utilisateur de *remonter* dans l'arborescence
SMNT	monte une nouvelle structure de données
REIN	réinitialise la session ou la connexion
QUIT	termine la session

- Les commandes de paramétrage des transferts :

PORT	déclare le point de connexion TCP du client (numéro attendu)
PASV	*renverse* le comportement du serveur sur la liaison de données ✓ Le serveur (ou le mode ou FTP) est dit alors passif car c'est le logiciel client qui initialise la connexion.
TYPE	contrôle le format de représentation utilisé pour les échanges
STRU	contrôle la structure des fichiers échangés
MODE	contrôle le mode d'échange des données

- Les commandes de service :

RETR	active le transfert du contenu d'un fichier à travers la liaison de données du serveur vers le client (téléchargement)
STOR	active le transfert du contenu d'un fichier à travers la liaison de données du client vers le serveur (dépose)
STOU	active une variante de la commande STOR en réalisant l'écriture dans un fichier de nom unique attribué par le serveur
APPE	active une variante de la commande STOR qui consiste à réaliser les écritures en fin du fichier s'il existe déjà
ALLO	alloue au serveur de la place en mémoire avant d'envoyer des données
REST	reprend un transfert sur un point de reprise
RNFR	renomme un fichier en précisant l'ancien nom (cette commande fonctionne avec RNTO)
RNTO	renomme un fichier en précisant le nouveau nom (cette commande fonctionne avec RNFR)
ABOR	demande l'arrêt immédiat d'un service en cours
DELE	supprime un fichier sur le serveur
RMD	supprime un répertoire sur le serveur
MKD	crée un répertoire sur le serveur
PWD	interroge le serveur pour connaître son répertoire courant
LIST	consulte la liste des fichiers d'un répertoire du serveur. Le format des données renvoyées n'est pas normalisé, car cette commande est libre d'implémentation.
NLST	consulte la liste des fichiers d'un répertoire du serveur. Le format des données est normalisé, ce qui permet à un programme client de réaliser des traitements automatiques de la réponse

SITE	réalise des opérations spécifiques au serveur et dont l'usage n'est pas considéré comme suffisamment *universel* pour apparaître dans la spécification du protocole
SYST	obtient des informations sur le système hôte
STAT	obtient des informations sur un transfert en cours (s'il y en a un), sur l'état du serveur ou sur un fichier particulier
HELP	obtient de l'aide sur les commandes implantées sur le serveur
NOOP	demande au serveur de répondre positivement

Un serveur minimal doit implémenter les commandes suivantes : TYPE, MODE, USER, QUIT, PORT, STRU, RETR, STOR et NOOP.

2.5 Réponses du serveur

Le serveur valide chaque commande du client par une réponse informant ce dernier du succès ou de l'échec de la requête. La réponse du serveur est constituée d'un code à trois chiffres suivi d'un texte explicatif. Tout comme les commandes, les réponses du serveur sont transmises sous formes de chaînes ASCII.
Par exemple, le serveur annonce la réussite d'un transfert de fichier par le texte suivant "226 Data Transfer Complete". La suite de caractères <CR><LF> figure également à la fin de chaque réponse.

Alors que le code-réponse à trois chiffres est prescrit par le protocole FTP, chaque serveur peut utiliser un texte explicatif librement défini. Cette règle ne connaît que quelques exceptions. En effet, les textes explicatifs des codes-réponse 110, 227 et 257 présentent un format imposé.

Les trois chiffres précédant le texte explicatif permettent théoriquement de coder mille réponses différentes à l'aide des codes 000 à 999. Le protocole FTP n'en utilise toutefois qu'une quarantaine.

Le premier chiffre d'une réponse indique si le serveur confirme (code 2xx) ou rejette (code 5xx) la commande FTP, ou s'il nécessite d'autres informations (code 3xx).

Le second chiffre indique la raison du rejet ou de l'acceptation d'une commande. Le serveur peut notamment rejeter une commande en raison d'une erreur de syntaxe (code x0x), confirmer l'établissement d'une connexion (code x2x), ou réclamer d'autres informations - c'est-à-dire les commandes suivantes (code x3x).

Le troisième chiffre sert à différencier les codes-réponse dont les deux premiers chiffres sont identiques. Ce chiffre n'est pas attribué selon un système établi, mais défini par le protocole FTP.

Le tableau suivant présente les codes-réponse les plus rencontrés assortis de leur interprétation la plus courante :

Code	Courte description du serveur	Explication
110	Marque de fichier (point de reprise)	Le serveur ne peut plus stocker un fichier donné. Il indique par cette réponse la position dans le fichier à partir de laquelle le transfert

		doit reprendre.
120	Serveur prêt dans *n* minutes	Le serveur accepte d'établir une connexion avec le client, mais étant surchargé, il ne peut autoriser aucune connexion d'ici *n* minutes.
125	Canal de données ouvert, début du transfert	Le fichier est transmis pour une liaison de données déjà ouverte.
150	Fichier prêt au transfert, tente d'ouvrir un canal	Le transfert du fichier/répertoire a été préparé. La liaison de données nécessaire est en cours d'établissement.
200	Commande OK	La dernière commande a pu être exécutée sans erreur.
202	Commande non requise par ce serveur et donc non implémentée	La commande définie par le protocole FTP n'est pas supportée par ce serveur.
211	*System Status* ou texte d'aide communiqué dans le reste de la réponse	Etat du serveur ou informations supplémentaires à propos d'une commande.
212	État d'un répertoire	Noms et attributs d'un répertoire.
213	État d'un fichier	Noms et attributs d'un fichier.
214	Texte d'aide dans le reste de la réponse	Aide concernant une commande (voir Help).
215	Identification du système d'exploitation du serveur	Communication du système d'exploitation du serveur (voir SYST).
220	Serveur prêt pour nouvel utilisateur	Le serveur est prêt à l'emploi, mais le client doit encore s'authentifier (voir 230).
225	Canal de données ouvert	Le canal de données a été ouvert. Les données sont transmises à la suite de cette réponse.
226	Transfert achevé, fermeture du canal de données	Le transfert des données est terminé.
227	Serveur passe en mode passif	Format fixe : *227 Entering Passive Mode* (i1,i2,i3,i4,ph,pl)
230	Nouvel utilisateur connecté	Le serveur met ses services à la disposition de l'utilisateur.
250	Opération de fichier exécutée	Dernière opération de fichier exécutée.
257	Répertoire créé. Format fixe : 257 <RépFichier>	Nouveau répertoire installé ou changement de répertoire.
331	Nom d'utilisateur reconnu, mot de passe requis	Pour achever l'identification d'un utilisateur, le serveur nécessite un mot de passe.
332	Login plus compte utilisateur	Le nom d'utilisateur et le mot de passe sont insuffisants. L'utilisateur nécessite également un compte utilisateur valable.
350	Autres informations requises pour exécuter l'opération de fichier demandée	Cette réponse est transmise par réaction à la commande RNFR pour signifier qu'une commande RNTO doit suivre.
421	Serveur/service non disponible.	Par ce message, le serveur referme de lui-même la session FTP. C'est notamment le cas lorsque l'administrateur de système du serveur arrête le système.
425	La liaison de données ne peut pas être établie	

426	Transfert interrompu, liaison de données refermée	Une erreur est survenue au cours du transfert des données.
450	Opération de fichier non exécutée	Un fichier/répertoire est inexistant ou l'utilisateur connecté ne peut pas y accéder.
451	Opération de fichier interrompue	Le serveur n'a pas pu exécuter une opération de fichier - c'est notamment le cas lorsqu'on tente de renommer un fichier sous un nom déjà existant.
452	Opération de fichier non exécutée	Le disque dur du serveur n'offre plus un espace suffisant pour accueillir le fichier qui devait être téléchargé.
500	Erreur de syntaxe, commande inconnue	La commande a été mal saisie.
501	Paramètre incorrect	L'un des paramètres de la commande est incorrect.
502	Commande non implémentée	La commande n'est plus supportée.
503	Commandes désordonnées	Cette erreur est provoquée lorsque la saisie du mot de passe précède celle du nom d'utilisateur.
504	Paramètre non autorisé	Le paramètre de transmission devant être défini par TYPE, MODE ou STRU n'est pas supporté par le serveur.
530	Utilisateur non inscrit	Le serveur ne reconnaît pas le nom d'utilisateur et le mot de passe.
532	Compte utilisateur requis pour l'enregistrement de fichiers	L'utilisateur n'est autorisé à sauvegarder des fichiers sur le serveur que s'il détient un compte sur le serveur FTP.
550	Action non exécutée	L'utilisateur a réclamé un fichier inexistant
551	Action non exécutée	
552	Action interrompue	L'utilisateur a dépassé l'espace disque qui lui était alloué sur le serveur pour enregistrer des fichiers.
553	Action non exécutée	Nom de fichier incorrect.
554	Action non exécutée, marque incorrecte	Le transfert de fichier ne peut pas être réactivé en raison d'une marque incorrecte.
555	Action non exécutée, TYPE ou STRU incorrect	Lors de la reprise du transfert de fichier, un fichier devait être étendu par APPE. Mais les valeurs TYPE et STRU du fichier devant être ajouté ne correspondent pas au fichier présent sur le serveur.

3. Types de sites

Les ressources en fichiers sont rassemblés sur les serveurs FTP en sites selon leur mode d'accès. Nous pouvons comparer cette organisation à celle des sites Web qui rassemblent les pages Web selon l'adresse URL d'accès.

Il existe deux types de sites classés selon le mode d'accès :

- **Site public à accès anonyme**,

- **Site privé à accès restreint** ou fermé.

Les deux types de sites diffèrent peu dans la procédure de connexion puisqu'il faut fournir au serveur dans les deux cas un **nom d'utilisateur** et un **mot de passe**.

3.1 Site privé

Un site est privé dans la mesure où ces deux identifiants sont confidentiels.

3.2 Site public

En ce qui concerne les sites publics, l'usage veut que tous les serveurs présents sur l'Internet mettent en œuvre un compte d'utilisateur invité ou anonyme dénommé **anonymous**. Le mot de passe de ce compte *anonymous* n'est pas mis en place, mais il est demandé de mettre son **adresse de courriel** (E-Mail) dans le champ mot de passe. Cet usage est écrit dans le ***RFC822***, et fait partie des nombreuses règles de la Nétiquette.
Depuis, les règles se sont assouplies à l'égard de ce mot de passe. Un mot de passe de la forme d'une adresse de courriel, comportant au moins le signe arobace, suffit. Certains logiciels serveurs ou administrateurs de serveurs autorisent même l'absence de mot de passe.
Le navigateur inclut dans son code les procédures d'authentification à l'aide du compte d'utilisateur anonyme sans que l'utilisateur ait à saisir un mot.

3.3 Organisation des sites publics

Les fichiers sont proposés sous forme d'arborescence de répertoires. Le répertoire de plus haut niveau (appelé ***répertoire racine***) est désigné par une barre oblique (/ comme sous UNIX…).

Ce répertoire contient généralement une demi-douzaine de sous-répertoires, mais un seul présente un intérêt sur les **sites publics** : il s'appelle **pub** (pour *public*). C'est dans ce dernier que nous trouvons l'ensemble des fichiers mis à disposition du public.

Sur les sites publics (mode anonyme), l'utilisateur ne peut en général envoyer de fichiers au serveur sur lequel il est connecté, bien que le protocole FTP le permette. L'administrateur du serveur peut cependant autoriser la dépose de fichiers sous un répertoire nommé conventionnellement **incoming** ou **upload**. Ces répertoires se situent au même niveau que le répertoire **pub** ou en dessous.

4. Serveurs

L'indépendance aux systèmes de fichiers fait de ce service FTP, à l'image des autres services du réseau Internet, un service hétérogène. Il s'affranchit du type de système d'exploitation aussi bien chez le client que chez le serveur. L'utilisateur sous Windows peut se connecter sur un serveur FTP sous UNIX, et vice-versa.

Le système d'exploitation Windows n'intègre pas dans son système d'exploitation de serveur FTP, à l'instar du système d'exploitation UNIX. Cependant, l'installation du serveur Web *Personal Web Server* sous Windows NT4 Workstation, ou *Internet Information Server* sous Windows NT4 Server ou Windows 2000, propose un service FTP minimum accolé au fonctionnement du serveur Web.

En conséquence, si les utilisateurs veulent installer un vrai serveur FTP avec toutes ses fonctionnalités, ils doivent chercher des programmes ou logiciels sur le réseau Internet.
En conséquence, dans la suite du document, l'auteur propose d'étudier :

- deux serveurs FTP sous Linux (ou sous UNIX) que sont **wu-ftpd** et **Proftpd**,
- un serveur FTP graphique sous Windows avec **Serv-U**.

4.1.1. Serveurs sous Windows

Les autres principaux serveurs FTP sous Windows sont :

- **BulletProofFTP Server** est le successeur du logiciel FTP *G6 FTP server*. Ce partagiciel en anglais en reprend toutes ses caractéristiques.

- *CrushFTP*, partagiciel en anglais, est un serveur FTP multiplateformes (Linux, Macintosh, OS/2, Unix). Il est écrit entièrement en Java et offre un interface d'administration à distance.

- *SurgeFTP* est un serveur FTP fonctionnant aussi bien sous Linux que sous Windows.

- *Crystal FTP 2000*, partagiciel en anglais, constitue un excellent choix pour opérer un site FTP car il combine plusieurs fonctions pour manipuler des fichiers et des répertoires dans un serveur FTP ainsi qu'un interface totalement configurable et convivial.

- *WS_FTP Server* est un Serveur FTP complet pour Windows.

- **War FTP Daemon**, gratuiciel en anglais, est un serveur FTP classé parmi les plus grands. Il présente une multitude de fonctionnalités très utiles et simples.

La liste ne s'arrête pas aux serveurs nommés ci-avant. D'autres serveurs sous Windows sont cités dans les sites spécialisés :

- Des logiciels payants : ArGoSoft FTP Server, Avirt Gateway, BisonWare FTP Server, Dragon Server, FtpMax, Vermillion FTP Daemon, WFTPD,
- Des logiciels gratuits : Gophers, Guild FTPd, Raiden FTPD.

4.1.2. Serveurs sous Linux

Dans les distributions Linux, les versions antérieures à la Mandrake 7.0 utilisaient **Bero-FTPd** (paquetage BeroFTPD-1.7.111-7mdk) téléchargeable à l'adresse ***ftp://beroftpd.unix.eu.org/pub/BeroFTPD***.
Entre les versions Mandrake 7.2 et 8.x, la préférence fut donné au serveur **wu-ftpd**.
Depuis la Mandrake 8.2, le serveur **ProFTPD** s'impose.

Le monde des logiciels libres bouge et d'autres serveurs FTP sont apparus plus axés sur la sécurité :

- **vsftpd**, un logiciel "sécurisé" disponible à l'adresse ***vsftpd.beasts.org*** (ou www.vsftpd.org)
- **pureftpd** dont la dernière version est *pureftpd-1.1.0-1.2.0.*, version disponible à l'adresse ***www.pureftpd.org***,
- **FTP4ALL (*www.ftp4all.de*)**, un logiciel européen qui ne nécessite pas de compte d'administrateur pour l'installer ou le compiler,

- **NcFTPd** (***www.ncftp.com***) qui est un serveur FTP à hautes performances, spécialement destiné aux sites à fort trafic et aux fournisseurs de services Internet.

5. Configurer un serveur en trois étapes

La configuration minimale d'un serveur passe par plusieurs étapes :

- L'identité du serveur ou du site,
- La gestion du nombre d'utilisateurs (en connexion simultanée, durée maximale d'inaction),
- L'attribution des droits d'accès.

5.1 Identité du serveur ou du site

Si le serveur publie un seul site, le site par défaut ou site principal, les problèmes d'identité ne se posent pas. Le serveur reprend l'adresse IP et le nom de domaine de la station qui accueille le service. Une alternative au nom de domaine de la station est celui d'un nom de domaine spécifique, commençant par le préfixe **ftp.*** et configuré dans le serveur DNS de rattachement.

Le numéro de port est le port standard. La question se pose si :

- la station informatique héberge plusieurs services Internet, et notamment le service Web ;
- le serveur héberge un (ou plusieurs) site(s) public(s) ou / et privés.

L'administrateur peut jouer avec les trois identifiants réseau suivants :

- l'**adresse IP** du site ou service,
- le numéro de **port TCP**,
- le **nom de domaine** du site visé.

Si l'administrateur veut jouer avec l'adresse IP, il peut en ajouter une ou plusieurs dites virtuelles (ou alias) à une seule carte réseau de la station informatique.
Il peut alors attribuer ou assigner (terme technique utilisé) une adresse IP :

- par service pour assurer une meilleure protection et administration des services,

ou

- à chaque site publié.

Si l'administrateur préfère manipuler les numéros de port, il est conseillé de dépasser la valeur de 2048. Ceux qui le désirent peuvent vérifier l'occupation des ports dans le RFC 1700 (valeurs normalisées par l'organisme de gestion du réseau Internet IANA).

C'est la 3ème solution que l'administrateur choisit de préférence pour :

- différencier les accès aux services (*www.nomdesite.com*, *ftp.nomdesite.com*, etc),
- différencier les accès aux sites (*ftp.serveur_internet.com*, *ftp.serveur_local.com*) selon l'origine des clients (réseau Internet, réseau local), ou la vocation du site.
    - Dans ce cas, l'administrateur mettra en œuvre des **serveurs virtuels** de la même façon que le fait le webmestre avec les serveurs virtuels Web.

5.2 Nombre d'utilisateurs connectés

L'administrateur d'un serveur FTP peut en outre fixer une limite à ce nombre d'utilisateurs connectés.
Si plusieurs services fonctionnent sur la même station, il est conseillé de bien répartir les connexions TCP disponibles.
Si le service Web est prioritaire, nous pouvons lui réserver le maximum de connexions (256 au plus sous Windows).
Si le serveur ne limite pas les connexions, mais le nombre d'utilisateurs, il ne faut pas oublier que le client FTP emploie deux connexions TCP pour la communication (ports 20 et 21 par défaut).

Si le nombre d'utilisateurs connectés est limité, l'administrateur doit aussi régler finement cette durée de connexion, ou plutôt la durée d'inaction avant déconnexion, afin de libérer la connexion au profit d'un autre utilisateur. La plupart des serveurs proposent cette fonctionnalité.

5.3 Attribution des droits d'accès

Cependant, l'essentiel du travail d'administration d'un serveur ne réside pas dans la répartition des ressources disponibles, mais dans l'attribution des droits d'accès.
L'administrateur d'un serveur doit avoir fait le point à l'avance des mesures de sécurité nécessaires.

Le serveur doit-il uniquement offrir un accès public ? Quels fichiers doivent être accessibles à tous les utilisateurs ? Qui est autorisé à déposer de nouvelles ressources ? Qui est autorisé à administrer le serveur à distance ?
Toutes les réponses à ces questions déterminent une structure, et une politique d'accès aux sites et aux ressources.

Il faut se rappeler que les noms d'utilisateurs et mots de passe circulent en clair sur le réseau si nous utilisons le protocole FTP standard. Etant donné que n'importe qui peut les récupérer et se substituer à un utilisateur reconnu par le serveur, l'administrateur a tendance à protéger plutôt les ressources que les accès.

Il peut le faire à deux niveaux :

- Au niveau du service (du serveur),
- Au niveau des fichiers eux-même.

Au premier niveau, l'administrateur a donc l'habitude de limiter les droits des ressources à :

- une simple lecture des fichiers pour permettre le téléchargement,
- un déplacement dans l'arborescence du site (commande *cd*),
- un listage des ressources à l'endroit où il se situe (commandes *dir* ou *ls* selon le système d'exploitation).

Par défaut, il interdit :

- le droit en écriture, droit qui autorise la dépose de ressources,
- la possibilité de créer des répertoires ou dossiers,
- la possibilité de supprimer les ressources, que ce soient des fichiers ou des répertoires.

En dernier lieu, il faut se rappeler que les droits des systèmes de fichiers l'emportent sur les droits. Pour plus de précaution, l'administrateur peut appliquer les mêmes verrous qu'au niveau du service.
Bien sûr, en dehors des ressources, l'administrateur peut indiquer un nom d'utilisateur autre que le nom *anonymous* des sites publics pour accéder aux sites privés.

Pour ajouter un mécanisme de protection supplémentaire, certains serveurs exigent l'indication d'un compte d'utilisateur en plus du nom et du mot de passe. Ce compte est indiqué par la commande cliente ACCT (Account). Les logiciels clients spécialisés prévoient un champ pour spécifier ce compte.

6. Compléments

6.1 Procédé FXP

Les actualités évoquent souvent le procédé **FXP** pour améliorer la facilité de transfert de fichiers entre deux serveurs FTP et un client. Du coup, ce procédé allant à l'encontre de la sécurité, les serveurs ont tendance à s'en protéger. Qu'en est-il exactement ?

L'utilisation de la technique FXP procure plusieurs avantages dont :

- l'augmentation de la sécurité en rendant relativement anonymes les connexions,
- l'emploi d'un compte FTP à ratio sans toucher à la limite de quota pour la dépose de fichiers,
- l'envoi à un tiers qui a installé un serveur FTP des fichiers d'un site FTP qu'il ne peut joindre (FTP privé par exemple),

Si le taux de transfert direct est trop faible, le logiciel FXP les transfère sur un site "relais" où le taux de transfert est meilleur. Dans un premier temps, nous commençons à balancer tout sur le site relais. Puis nous ouvrons une autre session FXP et nous récupérons les fichiers du site "relais". Comme la première connexion est de pur type FXP, cela n'ampute pas la bande passante.

6.2 Protocole Secure FTP

Le protocole FTP n'est pas sécurisé car les noms d'utilisateurs et mots de passe circulent en clair sur le réseau. Le protocole **S-FTP** ou **Secure FTP** essaie d'apporter une réponse à ce problème de plus en plus sensible.

7. Conclusion

Pour tous les autres renseignements concernant le service en lui-même, les outils de connexion à un site, les commandes, il est conseillé de se reporter au module « Services Internet ». Les notions en question sont considérées comme acquises, et donc ne seront pas reprises et détaillées dans ce document.
En conséquence, la lecture préalable du chapitre consacré au service FTP est nécessaire avant d'entreprendre la lecture de ce document, ou avant la mise en œuvre approfondie des serveurs exposés.

Chapitre 2 : Serveurs Linux

1. Serveurs sous UNIX

1.1. Démon ftpd standard

1.1.1. Mise en oeuvre

Les systèmes UNIX sont généralement livrés avec un service FTP par défaut. Le programme ou processus chargé de ce service est habituellement nommé **ftpd** ou **in.ftpd**.
Dans le cas du super-démon réseau **inetd**, il est activé en ajoutant la ligne suivante dans le fichier **inetd.conf**, fichier de configuration du super-démon.

```
ftp   stream  tcp    nowait  root    /usr/sbin/in.ftpd       in.ftpd
```

Ou

```
ftp   stream  tcp    nowait root    /usr/sbin/tcpd          in.ftpd
```

Cette ligne est toujours présente mais parfois commentée par la présence du caractère # en première colonne. Pour activer le service, il suffit de retirer ce caractère et redémarrer le super-démon *inetd*.

1.1.2. Site public

Ce programme peut suffire pour mettre en œuvre un site public et donc fournir un accès anonyme au service (cas des utilisateurs distants ne disposant pas d'un compte régulier sur le système UNIX).

Pour cela, il suffit d'effectuer quelques paramétrages :

1. tout d'abord vérifier, sinon créer le compte d'utilisateur **ftp** au niveau de la gestion des utilisateurs du système.
   - Cet utilisateur permet de s'identifier soit avec le nom *ftp*, soit avec le nom *anonymous*. En effet, la création du compte *anonymous* n'est pas nécessaire, car le compte d'utilisateur *ftp* est associé au niveau du programme à l'autre compte anonyme. Voici un exemple de configuration du compte extrait du fichier */etc/passwd* :

     ```
     ftp:x:uid:gid:FTP Daemon:/bebopalula/ftp:/bin/false
     ```

   - Le groupe d'appartenance de cet utilisateur est spécial (root par défaut). Le répertoire principal */bebopalula/ftp*, choisi selon la convenance de l'administrateur du site, sera la racine protégée de l'arborescence publique (via la commande **chroot()**). D'autre part, il est recommandé d'attribuer au compte *ftp* un mot de passe *impossible* ainsi qu'un interpréteur de commandes leurre (ici /bin/false), de façon à rendre ce compte inutilisable de l'extérieur.

2. créer un alias de courrier permettant à tout un chacun de correspondre avec l'administrateur du site.

- Il suffit pour cela de rajouter la ligne qui suit dans le fichier système *aliases*, avant d'exécuter la commande **newaliases** :

ftp-admin: utilisateur@machine

3. créer le répertoire racine de l'arborescence anonyme (ici */bebopalula/ftp*), en rendre propriétaire le super utilisateur (*root*), puis protéger l'accès à ce répertoire en positionnant les droits *r-xr-xr-x* (chmod 555 */bebopalula/ftp*)

4. créer un certain nombre de sous-répertoires nécessaires au bon fonctionnement du service pour le confiner dans cette partie du système de fichiers :

   - **.../ftp/bin**
   en rendre propriétaire le super utilisateur, positionner les droits *--x--x--x* (chmod 111 …) et y déposer un copie de l'exécutable */bin/ls* bénéficiant des mêmes propriétaires et droits.

   - **.../ftp/usr/lib**
   en rendre propriétaire le super utilisateur, positionner les droits *--x--x--x* et y déposer une copie des bibliothèques partagées nécessaires au bon fonctionnement des diverses commandes du répertoire *bin* ainsi que du démon *ftpd* lui-même.
   Pour Solaris, il est recommandé d'y placer les copies de bibliothèques suivantes : ld.so.1, libc.so.1, libdl.so.1, libmp.so.1, libnsl.so.1, libsocket.so.1, nss_compat.so.1, nss_dns.so.1, nss_files.so.1, nss_nis.so.1, nss_nisplus.so.1, nss_xfn.so.1, straddr.so et straddr.so.2.

   - **.../ftp/etc**
   en rendre propriétaire le super utilisateur, positionner les droits --x--x--x et y déposer une copie des fichiers */etc/passwd* et */etc/group* (le système Solaris recommande aussi une copie du fichier */etc/netconfig*).
   Ces fichiers doivent être aussi propriété du super utilisateur et protégés par les droits *r--r--r--*.
   Une précaution supplémentaire s'impose pour *maquiller* les véritables utilisateurs du système : il faut purger aussi les fichiers *.../ftp/etc/passwd* et *.../ftp/etc/group* des identités superflues.

   - **.../ftp/pub**
   Ce répertoire est la *véritable* racine de l'arborescence du site public. À partir d'ici, il revient à l'administrateur du site de choisir les propriétés et accès des différents fichiers et répertoires. Le programme utilisera comme propriétaire effectif le compte utilisateur *ftp*.

   - **.../ftp/dev**
   Ce répertoire n'est pas obligatoire, mais sur certains systèmes, il est nécessaire d'y faire figurer quelques périphériques. Solaris recommande d'y placer une *copie* (obtenue en utilisant *mknod)* des périphériques suivants : /dev/zero, /dev/tcp, /dev/udp et /dev/ticotsord.

Bien entendu, d'autres possibilités sont offertes. Il est possible d'autoriser ou d'interdire de déposer (upload) des fichiers, de consulter librement certains répertoires, voire même de créer des utilisateurs fictifs.

1.1.3. Fichier *ftpusers*

Le fichier **/etc/ftpusers** est utilisé afin d' **interdire tout accès au service FTP aux utilisateurs système** sensibles. Il suffit d'y faire figurer les comptes désirés.

```
root
daemon
bin
sys
adm
lp
smtp
uucp
nuucp
listen
nobody
noaccess
```

1.2. Présentation de Wu-Ftpd et ProFtpd

Les deux principaux serveurs FTP sous Linux sont actuellement :

- Wu-Ftpd
- ProFtpd

Si nous sommes habitués à gérer le serveur Web Apache, nous n'éprouverons aucune difficulté à configurer le serveur **Proftpd**, car sa configuration ressemble beaucoup à celle du serveur Apache. De plus, il intègre des outils de diagnostics très utiles (ratios, limitations de bande passante ...). Cependant, il semble gourmand en mémoire. Nous l'employons pour monter des services FTP privés.

Le logiciel **Wu-ftpd** quand à lui est très bien, mais moins maniable que le serveur *Proftpd*. C'est un produit de remplacement du service FTP standard (**in-ftpd**). Il est très apprécié pour fournir un service FTP public à haute vitesse qui incluse en outre des capacités multimédia.
Le logiciel *Wu-ftpd* permet de réaliser des contrôles d'accès plus fins en **classant les utilisateurs anonymes** selon divers critères et en autorisant ou interdisant certaines fonctionnalités à partir de ces classes. D'autre part, des fichiers journaux traçant les sessions de façon évoluée sont possibles, ainsi que des **compressions** ou décompressions **à la volée**.
Cependant, quelques **trous de sécurité** ont suffi pour altérer sa notoriété. Pour pallier cette limitation, il suffit de télécharger la dernière version.

1.3. Procédures d'installation

1.3.1. Installation proprement dite

Il existe deux façons de les installer :

- à partir du cédérom de distribution Linux,

- en téléchargeant depuis le site associé au serveur.

a. Cédérom

Dans le cas du cédérom, les logiciels apparaissent sous la forme de paquetages (ou packages). En général, il s'agit de paquetages portant l'extension *rpm*.
Sur les consoles non graphiques, nous utilisons les commandes du même nom *rpm* :

```
rpm -q wu-ftpd                                  # pour interroger
      wu-ftpd-2.4.2b12-6
```

Contrairement à l'exemple ci-dessus, il se peut qu'il y ait plusieurs paquetages à installer où le nom du logiciel reste l'élément commun. Dans ce cas, la dépendance entre ces différents paquetages peut imposer un ordre d'installation.

```
rpm –ivh nom_du_serveur-mdk-i586.rpm            # pour installer avec les dépendances
```

Sur les consoles graphiques, plusieurs programmes sont mis à notre disposition pour réaliser cette installation : ***kpackage*** (environnement graphique KDE) ou ***gnorpm*** (environnement graphique Gnome) par exemple.

b. Téléchargement

Dans le cas du téléchargement, le répertoire */tmp* peut servir à décompresser le fichier **.tar.gz* à l'aide de la commande :

```
tar -zxvf fichier.tar.gz
```

Une fois cette opération effectuée, il faut se déplacer dans le répertoire créé lors de l'extraction du fichier (commande « cd *répertoire* ») afin de passer successivement les commandes de configuration, de compilation et d'installation :

```
1. ./configure --prefix=/usr --sysconfdir=/répertoire
   make
2. make install

3.
```

Les paramètres *prefix* et *sysconfdir* permettent de dire respectivement ou installer et ou aller chercher les fichiers de configuration.
L'absence d'erreur signalée lors des trois opérations précédentes signifie que le serveur est bien installé.

1.3.2. Vérifications post-installation

Avant de commencer, il faut décider du mode de fonctionnement du serveur :

- soit le faire gérer par le super service réseau (processus démon) du système :
  - **inetd** pour les distributions UNIX commerciales ou anciennes distributions Linux
  - **xinetd** pour les distributions Linux récentes
- soit de manière autonome

Le choix du service réseau est valable dans le cas d'un service FTP occasionnel. Le fonctionnement autonome est à préférer dans le cas de la mise ne œuvre d'un FTP Internet opérationnel.

En fonction du choix effectué, les vérifications ne seront pas les mêmes.
Dans le premier cas, nous contrôlons que le port du service FTP pointe bien sur le serveur FTP de notre choix.

En ce qui concerne le démon **inetd**, le fichier **/etc/inetd.conf** nous renseigne par les lignes suivantes :

```
ftp stream tcp nowait root /usr/sbin/tcpd wu.ftpd-2.6 –l –a
#ftp stream tcp nowait root /usr/sbin/tcpd proftpd
```

Les deux lignes permettent de pointer vers le serveur de son choix. Dans le cas présent, le logiciel *wu.ftpd* a été choisi (absence du caractère # de commentaire en début de ligne).
Le programme est invoqué avec les options *-l* et *-a* par inetd :

- l'option *-l* indique que tous les accès ftp seront inscrits dans le fichier journal */var/log/message* par le processus *syslog,*
- l'option *-a* impose d'utiliser le fichier de paramétrage */etc/ftpaccess* du serveur *wu-ftpd.*

En ce qui concerne le démon **xinetd**, le fichier **/etc/xinetd.conf** nous renseigne par les lignes suivantes :

```
…
includedir /etc/xinetd.d/
```

Dans le répertoire en question, il existe un fichier texte de configuration associé au serveur choisi, par exemple **proftpd-xinetd** :

```
# default: off
# description: proftpd server, xinetd version. \
# Don't run the standalone version if you run \
# this!

service ftp
{
        disable                 = yes
        socket_type             = stream
        wait                    = no
        user                    = root
        server                  = /usr/sbin/in.ftpd
        log_on_success          += DURATION USERID
        log_on_failure          += USERID
        nice                    = 10
}
```

Le service FTP par défaut (in.ftpd) est désactivé (1ère ligne de commentaire et ligne disable=yes).

Si nous choisissons d'installer le serveur *Proftpd*, il faut savoir que celui-ci fonctionne par défaut de manière autonome (ou *standalone*).

Dans le cas d'un **fonctionnement autonome** du serveur, il n'existe, en conséquence, **pas de vérification** à effectuer.
Si le fichier de configuration du service réseau est modifié, il faut relancer le service pour que les modifications soient prises en compte, notamment à l'aide de la commande :

```
killall –HUP inetd OU xinetd
```

1.3.3. Lancer le serveur

Pour lancer le serveur, il suffit de rendre sous */etc/rc.d/init.d*, puis de lancer le script associé portant le même nom que le logiciel utilisé :

```
./proftpd start
        Lancement du serveur FTP (proftpd) :      [OK]
```

Pour arrêter le serveur, nous utilisons la commande : *./proftpd stop*.
Il peut arriver que nous ne sachions plus si le serveur est démarré ou pas. Nous pouvons simplement interroger l'état du serveur : *./proftpd status*.

1.3.4. Tester le serveur

Maintenant, il reste à tester localement si le serveur choisi répond bien en simulant la connexion d'un client au serveur. Il suffit de lancer la commande :

```
$ ftp localhost
```

Si le serveur affiche un message de bienvenue, cela signifie que le serveur fonctionne bien !
L'absence de message signifie que le serveur présente des dysfonctionnements. Dans ce cas, il est conseillé de faire apparaître, dans une autre console, la fin du fichier log **messages**, par la commande :

```
tail -f /var/log/messages
```

Les dernières lignes de ce fichier peuvent dévoiler la cause des problèmes rencontrés.

1.4. Environnement de fonctionnement

1.4.1. Autres programmes complémentaires

Selon les types d'UNIX ou LINUX, l'administrateur peut mettre en œuvre les programmes suivants :

- ***/usr/bin/ftpcount*** pour compter le nombre d'utilisateurs connectés au serveur à un instant "t"
- ***/usr/bin/ftpwho*** pour indiquer quel utilisateur est connecté à l'instant "t"
- ***/usr/sbin/ftpshut*** pour écrire un message d'arrêt du serveur aux clients

- ***/usr/sbin/ftprestart*** pour redémarrer le service après l'utilisation de la commande précédente
- ***/usr/sbin/xferstats*** pour lire le fichier ***/var/log/xferlog***, fichier journal qui recueille les statistiques d'accès au serveur FTP. Le format du fichier est expliqué par le format de l'aide en ligne du système (commande *man xferlog*).

Dans le cas du serveur *wu-ftpd*, les statistiques sont activées si la directive *log* du fichier */etc/ftpaccess* est bien mise en œuvre, à l'image de l'exemple qui suit.

```
log transfers guest outbound
```

Cet exemple indique que le serveur *wu-ftpd* veut tracer les transferts en sortie (outbound) du serveur (téléchargements) pour le seul type d'utilisateurs *guest*.

1.4.2. Fichier */etc/ftphosts*

Il existe un fichier **ftphosts** pour filtrer les clients du serveur à la fois par leur nom d'utilisateur, et par leur adresse IP ou nom de domaine. La syntaxe est la suivante :

```
allow <nom d'utilisateur> <addrglob> [<addrglob> ...]
deny < nom d'utilisateur> <addrglob> [<addrglob> ...]
```

Voici l'exemple d'un tel fichier où l'on autorise tous les utilisateurs se connectant depuis les stations dont le nom de domaine est *esat.terre.def*, où l'on interdit tous les utilisateurs en provenance des stations du réseau *131.211.32.0* !

```
allow  *  *.esat.terre.def
deny   *  131.211.32.*
```

Le caractère « * » est utilisable en tant que joker. Les adresses peuvent être précédées du caractère « ! » pour indiquer une négation.
Une ligne accepte une seule directive *allow* ou *deny*. Par contre, le fichier accepte plusieurs directives *allow* et *deny*. Si nous utilisons les deux ensemble, l'ordre peut être significatif.

```
allow toto *
deny toto svr.esat.terre.def
```

L'exemple précédent autorise l'utilisateur *toto* à se connecter depuis n'importe quelle station sauf celle dont l'adresse est *svr.esat.terre.def*.
L'exemple suivant autorise l'utilisateur *toto* à se connecter depuis n'importe quel station.

```
deny toto ga.bu.zo.meu
allow toto *
```

1.4.3. Fichier */etc/ftpconversions*

Cette fonctionnalité est assez méconnue car le format du fichier reste assez hermétique. Il permet aux utilisateurs de réaliser à la volée des archivages (tar), des compressions (gzip ou compress) ou décompressions sur les données à télécharger.

Plutôt que d'offrir les mêmes données compressées sous divers formats afin que les utilisateurs économisent du temps lors des transferts, il est possible de n'offrir qu'un exemplaire de base aux clients tout en permettant de choisir leur format préféré.
Ce fichier contient autant de lignes désirées selon le format suivant.

pa:sa:pc:sc:cmd:type:options:description

pa	désigne un préfixe à rechercher dans le nom du fichier du serveur.
sa	désigne un suffixe à rechercher dans le nom du fichier du serveur.
pc	désigne un préfixe à rechercher dans le nom du fichier indiqué par le client.
sc	désigne un suffixe à rechercher dans le nom du fichier indiqué par le client.
cmd	désigne l'action à réaliser par le serveur (lire plus bas).
type	désigne le type des objets sur lesquels l'action est réalisable. Les valeurs possibles sont une combinaison de T_REG, T_DIR et T_ASCII.
options	désigne de façon générique l'action effectivement réalisée. Les valeurs possibles sont une combinaison de O_COMPRESS, O_TAR, O_UNCOMPRESS et O_UNTAR.
description	affiche un message informationnel à des fins de traçage.

Voici un exemple :

```
:.gz: :    :/bin/gzip -cd %s:T_REG|T_ASCII:O_UNCOMPRESS:GUNZIP
:    : :.gz:/bin/gzip -9 -c %s:T_REG:O_COMPRESS:GZIP
:    : :.tar:/bin/tar -c -f - %s:T_REG|T_DIR:O_TAR:TAR
:    : :.tar.gz:/bin/tar -c -z -f - %s:T_REG|T_DIR:O_COMPRESS|O_TAR:TAR+GZIP
```

La première ligne indique que si l'utilisateur demande le transfert d'un fichier *fic* alors que le serveur détient uniquement une version compressée au format *fic**.gz***, le serveur décompressera ce dernier à la volée.
La deuxième ligne permet une compression automatique d'un fichier archive lors du transfert.
La troisième ligne s'applique sur des fichiers ou des répertoires que le client veut télécharger au format ***tar***.
La quatrième ligne s'applique sur des fichiers ou des répertoires que le client veut télécharger au format *tar* compressé.

Attention :
Pour que cela fonctionne, il ne faut pas oublier de placer dans le répertoire */bin* de l'arborescence anonyme les commandes utilisées, ainsi que les bibliothèques dynamiques éventuellement nécessaires à leur bon fonctionnement.

1.4.4. Fichier */etc/ftpgroups*

Ce fichier contient une liste de pseudo-groupes associés à de véritables groupes système. Les groupes sont protégés par un mot de passe. Leur utilisation permet au serveur FTP d'utiliser les droits d'accès du groupe système pour accéder aux fichiers et répertoires déposés sur le site.

Chaque ligne de ce fichier est au format suivant :

- *pseudo-groupe* est l'identification à utiliser avec la commande *SITE GROUP pseudo-groupe* (voir la commande *man ftpaccess* pour le format)

- *mot de passe* est le mot de passe **crypté** à utiliser avec la commande *SITE GPASS mot de passe*
- *groupe* est l'identité système du groupe utilisé par le programme pour obtenir des droits différents

pseudo-groupe:mot de passe:groupe

Voici un exemple de fichier :

test:mot de passe crypté:hendrix

La modification de ce fichier peut-être réalisée avec la commande *privatepw* (serveur wu-ftpd).

1.4.5. Fichier */etc/ftpservers*

Ce fichier de configuration permet de gérer des fichiers de configuration de **serveurs virtuels** similaires à ce que propose le serveur Web Apache. Il indique au serveur *ftpd* quel fichier de configuration utiliser pour tel ou tel serveur virtuel.
Quand le support des serveurs virtuels est installé, le logiciel *wu-ftpd* a la capacité d'utiliser des fichiers séparés.

Les fichiers de configuration sont souvent placés dans un répertoire propre à chaque serveur virtuel.
La syntaxe du fichier présente deux champs par ligne.

ipaddr/hostname directory-containing-configuration-files

Dans l'exemple ci-dessous, les deux premières lignes orientent les connexions du client vers le site FTP identifié soit par l'adresse IP, soit par le nom de domaine équivalent. La troisième ligne oriente le client vers un autre site FTP. La quatrième ligne oriente le client vers le fichier de configuration principal.

```
10.196.145.10          /etc/ftpd/ftpaccess.somedomain/
ftp.some.domain        /etc/ftpd/ftpaccess.somedomain/
10.196.145.200         /etc/ftpd/ftpaccess.someotherdomain/
some.domain            INTERNAL
```

1.4.6. Autres fichiers complémentaires

Il existe, selon les versions de LINUX, d'autres fichiers en rapport avec ce service :

- */etc/logrotate.d/ftpd* (script de rotation des fichiers journaux périodiques),
- */etc/pam.d/ftp* (privilèges d'utilisateurs FTP privés).

2. Serveur wu-ftpd

2.1. Sites de référence

Le site du groupe de développeurs indépendants associé au logiciel serveur, produit à l'origine par l'université de Washington de St-Louis, est à l'adresse ***www.wu-ftpd.org***, celui des archives wu-ftpd est à l'adresse ***wuarchive.wustl.edu*** et celui du groupe de développement du serveur à l'adresse ***www.academ.com/academ/wu-ftpd***.

Pour l'aide en ligne sous Linux, l'utilisateur peut consulter les différents fichiers situés sous */usr/doc/wu-ftpd-2.6.0/HOWTO*.

2.2. Plate-formes et versions

La dernière version connue est *wu-ftpd-2.6.2-9*.

Les plate-formes matérielles et systèmes d'exploitation supportées sont :

- Plate-formes Intel
  - BSD/OS 1.1, and 3.1
  - FreeBSD 2.2.6
  - Linux 1.2.X and 2.X
  - SCO OpenServer 5.x
  - SCO UnixWare 2.1
  - Solaris 2.4, 2.5.1 and 2.6
- Plate-formes Sun Sparc
  - Solaris 2.6
  - Solaris 2.5.1
  - SunOS 4.1.4

2.3. Arborescence

L'arborescence reste simple à expliciter :

- Le répertoire de publication sous /home/ftp,
- Les fichiers de configuration sous /etc,
- Le script de démarrage sous /etc/rc.d(/init.d).

Normalement, lors de l'installation du serveur *wuftpd*, des répertoires d'environnement ont été créés dans **/home/ftp**. Il s'agit des répertoires *bin*, *etc* et *lib*.

2.4. Service public

Dans un premier temps, nous expliquons comment mettre en œuvre un serveur public, acceptant les connexions anonymes.

Le compte d'utilisateur système **ftp** est utilisé dans le cadre du fonctionnement interne du serveur. La racine du site public est fréquemment localisée sous */home/ftp*. Les répertoires et fichiers des niveaux inférieurs appartiennent par défaut à l'utilisateur *root* du groupe *ftp*.

Le principal fichier de configuration est **/etc/ftpaccess**. Il contient de très nombreuses directives classées selon les catégories suivantes : contrôles d'accès, informations, traces, ratios, divers et permissions.

Trois types d'utilisateurs peuvent être distingués :

real Ce sont des utilisateurs possédant un compte UNIX sur la station serveur (compte déclaré dans le fichier de gestion des utilisateurs */etc/passwd*).
guest Ce sont des utilisateurs possédant un compte UNIX sur la station mais tout autant déclarés dans le serveur FTP en vue d'une utilisation privée.
anonymous Ce sont les véritables utilisateurs invités ou anonymes du service.

2.4.1. Téléchargement

Voici un contenu type du fichier *ftpaccess* qui assure un téléchargement de fichiers depuis le répertoire pub situé à la racine du site !

```
1    class all real,anonymous *
2    limit all 10 Any /etc/ftpmsg.dead

3    readme README* login
     readme README* cwd=*
     shutdown /etc/shutmsg
     message /welcome.msg login
     message .message cwd=*

4    passwd -check rfc822
```

1. Par défaut, le serveur déclare une classe d'utilisateurs **all** qui rassemble les deux types d'utilisateurs **real** et **anonymous**.
   L'étoile qui termine la ligne signifie que le service est accessible à cette classe depuis n'importe quelle station, quelle que soit son réseau d'appartenance.

2. Cette ligne limite le nombre de clients à 10, quelle que soit leur classe d'appartenance (all), quelle que soit la date et l'heure de connexion (Any). Le message indiqué leur est envoyé en cas de dépassement de seuil.

3. Ces lignes définissent le nom des fichiers qui seront consultés lorsque le client se connecte (login), passe dans un répertoire (cwd), à l'arrêt du service (shutdown). Les fichiers désignés doivent être présents dans le site pour voir leur contenu s'afficher. Les fichiers concernant les répertoires doivent être actualisés.
   Si les fichiers indiqués sont absents, ils n'altèrent pas le fonctionnement du serveur.

4. Il faut aussi spécifier le type de mot de passe. Conventionnellement, un mot de passe reprenant la syntaxe d'une adresse de courrier électronique est attendu (RFC 822).

Une fois le fichier *ftpaccess* modifié, avant de tester le serveur, il ne faut pas oublier de toujours redémarrer le service pour que la nouvelle configuration soit prise en compte.

2.4.2. Dépose

Pour permettre la dépose de fichiers par le client sur le serveur, l'administrateur de site public met en œuvre la directive **upload**.

Il crée tout d'abord le répertoire sous lequel les clients seront autorisés à déposer leurs fichiers. Ce dernier peut s'appeler aussi bien **Incoming** que **upload**. Il peut être créé à la racine du serveur tout comme sous le répertoire *pub*.
La syntaxe et l'emploi de cette directive étant relativement compliquée, il est donc recommandé de lire la documentation accompagnant le serveur.

```
upload [absolute|relative] [class=classe] racine répertoire \
   yes|no [propriétaire groupe mode [dirs|nodirs] [mode répertoire]]
```

Dans le fichier de configuration ***/etc/ftpaccess***, il faut rajouter aux moins deux lignes commençant par la directive **upload** :

```
upload      /home/ftp    *                no
upload      /home/ftp    /Incoming   yes  root  ftp   0666
```

La première ligne interdit la dépose où que ce soit sur le site, et notamment depuis la racine du serveur. La deuxième définit le chemin d'accès et le nom du répertoire de dépose (upload). Les fichiers déposés par le client appartiendront au propriétaire *root* et au groupe *ftp* avec les droits suivants [0666].
La ligne peut être complétée par une option **nodirs** pour interdire au client la création de sous-répertoires.
La syntaxe montre que nous pouvons employer aussi bien des chemins relatifs qu'absolus, et préciser la classe d'utilisateurs autorisés. Dans le cas où la création de sous-répertoires est admise, nous pouvons indiquer les droits à leur création.

Messages d'erreur :

a) Permission denied (upload) — Il doit persister une erreur dans le contenu du fichier *ftpaccess*.
b) Permission denied — L'administrateur doit modifier les droits des fichiers côté serveur sur le répertoire *Incoming*.
La commande **chmod g+w Incoming** autorise à coup sûr la dépose.

Voilà qui termine la configuration de notre serveur *wu-ftpd* en accès anonyme.

2.5. Service privé

Nous allons maintenant configurer le serveur *wu-ftpd* pour pouvoir l'utiliser avec des vrais comptes d'utilisateurs système gérés exclusivement par le service FTP.

Un vrai compte *ftp* consiste à définir :

- des comptes spécifiques requérant une authentification de base (*login*, *password*),
- un répertoire privé

- une structure permettant de confiner l'utilisateur sur son répertoire personnel, et empêchant les autres utilisateurs de se promener dans son arborescence.

2.5.1. Groupe invité

Nous allons donc utiliser les propriétés d'un groupe d'utilisateurs spécifique au service FTP, celui affecté au paramètre *guestgroup* dans le fichier *ftpaccess* : *ftpwww*.

```
guestgroup ftpwww
```

Puis, nous créons, au niveau du système UNIX, le group ad hoc par la ligne de commande :

```
groupadd ftpwww
```

Nous allons maintenant rajouter nos invités au niveau de la directive *class all* par la valeur **guest** :

```
class all real,guest *
```

Les utilisateurs "invités" (ou guest) se connectent sur le serveur avec un mot de passe dans un espace disque restreint ! L'utilisateur invité doit donc posséder un vrai compte sur le serveur, donc une entrée dans le ficher */etc/passwd*.
La seule différence avec la classe "real" précédemment indiquée provient du fait que les utilisateurs invités se retrouvent après connexion dans un espace disque limité. Le serveur effectue une commande **chroot** qui modifie la racine du site. Comme pour un accès anonyme, l'utilisateur invité se retrouve connecté à partir d'une racine "/" virtuelle. Cette racine nouvelle est définie dans le champs "home directory" de l'entrée de l'utilisateur dans le fichier */etc/passwd*.

2.5.2. Utilisateurs invités

Au niveau des utilisateurs, nous devons créer un véritable utilisateur invité ou *guest* dans le fichier *ftpaccess* et créer un véritable compte système associé.

Pour cela, il faut d'abord activer la directive *guestuser* dans le fichier *ftpaccess,* par exemple :

```
guestuser webmaster
```

Le compte système créé en parallèle est rattaché au groupe précédent (option –G).

```
useradd webmaster –G ftpwww
```

Un mot de passe lui est défini.

```
passwd webmaster
```

Notre premier utilisateur est donc un webmestre. Le site Web qu'il doit modifier se trouve dans le répertoire *home/www*. Il faut donc que cet utilisateur *webmaster* ne puisse pas remonter de son répertoire initial.
Pour cela, nous modifions le fichier */etc/passwd* en changeant le champ *home directory* comme ceci :

Avant	Après
/home/webmaster	**/home/webmaster/./www**

Le champ *home directory* est désormais divisé en deux parties séparées par un point "/./". La première partie indique le déplacement de la racine du site ftp (chroot). La deuxième partie représente le répertoire d'accueil de l'utilisateur invité !
Pour que l'utilisateur soit limité au seul service FTP (pas de connexion Telnet par exemple), il faut mettre en place un shell de connexion particulier ! Nous proposons un petit shell particulier appelé */etc/ftponly* qui sera lancé comme shell lors d'une connexion. Il est nécessaire de placer le nom de ce shell dans le fichier */etc/shells*.

Voici un exemple de fichier */etc/ftponly* :

```
$!/bin/sh
# ftponly shell
#
trap "/bin/echo Sorry; exit 0" 1 2 3 4 5 6 7 10 15
#
IFS=""
Admin=libes@com.univ-mrs.fr
System=`/bin/hostname`
#
/bin/echo
/bin/echo "*****************************************************************"
/bin/echo "    You are NOT allowed interactive access to $System."
/bin/echo
/bin/echo "     User accounts are restricted to ftp and web access."
/bin/echo
/bin/echo "  Direct questions concerning this policy to $Admin."
/bin/echo "*****************************************************************"
/bin/echo
```

La ligne de déclaration de l'utilisateur *webmaster* dans le fichier */etc/passwd* devient :

```
webmaster:x:1031:200: Webmestre :/home/ftp/./www:/etc/ftponly
```

Maintenant, il reste à tester le service FTP. Après la fourniture du mot de passe, si le serveur affiche quelque chose comme *Access restriction Apply* dans la bannière de bienvenue, cela signifie que l'utilisateur connecté fait partie du groupe d'invités *guestgroup*.

Incident possible :
Sous l'invite *ftp*, si l'utilisateur tape la commande *ls*, rien n'apparaît. Ceci s'explique par le fait que cette commande n'est pas intégrée à l'environnement du service FTP sous */home/webmaster*.
Pour vérifier qu'une commande fonctionne bien après avoir fait un changement de racine virtuel, on peut tester la commande suivante :

```
chroot ~ftp /bin/ls
```

Si la commande *ls* affiche quelque chose, cela signifie qu'elle reste disponible après avoir fait le changement de racine *~ftp*.

Pour pallier ce dysfonctionnement, la solution consiste à **copier les répertoires *bin*, *etc* et *lib*** situés à la racine du serveur *wu-ftpd* (*/home/ftp*) dans le répertoire */home/webmaster*.

2.6. Administration

Toutes ces options peuvent être rajoutées dans le fichier */etc/ftpaccess*. Elles sont rangées par catégorie : contrôles d'accès, informations, traces, ratios, divers et permissions.

2.6.1. Contrôles d'accès

La directive **class** permet de définir une classe d'utilisateurs en combinant des types (real, guest ou anonymous) et des localisations. Voici quelques exemples qui respectent la syntaxe :

```
class classe liste-de-type adresse [adresse ...]
```

```
class local    real,anonymous     *.orleans.orstom.fr
class remote real                 *.orstom.fr *.rio.net *.univ-orleans.fr
class all      guest,anonymous    *
```

La classe *all* prend en compte les autres cas.

- Pour interdire l'accès à des clients de tel réseau, ou dont le nom de domaine n'est pas répertorié dans le serveur DNS, nous pouvons insérer les lignes suivantes. L'interdiction n'empêche pas d'afficher le message contenu dans le fichier indiqué.

```
deny *.pepito.fr     /.ftpdeny_message
deny !nameserved /usr/local/etc/msgs/msg.noname
```

Voici un exemple de message indiqué :

```
Désolé, ce service est réservé aux utilisateurs du … !
Vous vous êtes connectés depuis %R.
Pour toute réclamation, contactez %E.
```

- Afin de limiter le nombre de personnes connectées simultanément sur le serveur, la directive **limit** ainsi configurée exprime le fait que, toutes classes confondues, dix utilisateurs maximum sont acceptés.
  Si une onzième personne se connecte, elle verra s'afficher le message contenu par le fichier */etc/ftpmsg.dead*.

```
limit all      10 Any                    /etc/ftpmsg.dead
limit guest 0   SaSu|Any2000-0600 /usr/local/etc/msgs/msg.timeout
```

La deuxième ligne mentionne l'interdiction du site aux invités, le samedi et dimanche, ou entre 20h00 et 06h00, et leur envoie le message indiqué en cas d'essai de connexion. Voici un exemple de fichier :

```
Bonjour %U,

Il nous semble qu'en ce moment, vous êtes de trop. En effet les règles en vigueur
n'autorisent pas plus de %M utilisateurs simultanément à ce moment précis de la
journée %T.

Veuillez réessayer plus tard.

Pour toute réclamation, contactez %E.
```

- La directive **loginfails** permet de limiter le nombre d'essais de connexions, par exemple à trois dans l'exemple qui suit (nombre conventionnel).

```
loginfails 3
```

- La directive **noretrieve** permet de rendre impossible toute tentative d'accès à certains fichiers ou répertoires de l'archive. Voici un exemple où l'on interdit l'accès à tout fichier situé sous les répertoires */etc, /usr, /dev, /bin* et */lib* ainsi qu'aux fichiers portant le nom *core* où qu'ils se trouvent.

```
noretrieve /etc /usr /dev /bin /lib core
```

- Les autres directives relatives au même sujet sont : private, autogroup, realgroup, realuser, nice, defumask, tcpwindow, keepalive, timeout accept, timeout connect, timeout data, timeout idle, timeout maxidle, timeout RFC931, file-limit, data-limit, guestserver, allow-retrieve.

2.6.2. Délivrance d'informations

- La directive **banner** sert à accueillir le client avant la procédure d'identification, par l'affichage d'une bannière.

```
banner /home/ftp/banner.txt
```

Voici un exemple de fichier :

```
Bienvenue sur le serveur FTP de notre organisation.
```

Attention :

Certains clients FTP ne supportent pas la présence de messages de plus d'une ligne.

- La directive **email** indique l'adresse de courriel de l'administrateur du serveur, en cas de problème de fonctionnement de ce dernier.

```
email webmaster@ze-linux.com
```

- La directive **greeting** permet de contrôler le message envoyé lors de la connexion.

```
greeting full|brief|terse
greeting text message
```

L'option *full* employée par défaut mais non recommandée fait apparaître le nom de la machine ainsi que la version du serveur. L'option *brief* ne fait apparaître que le nom de la station. Quant à la troisième forme, elle ne fait apparaître qu'un message minimal. La quatrième forme permet de préciser le contenu du message.

- La directive **message** permet de définir des messages à afficher lorsque certains évènements se produisent. Sa syntaxe est :

message chemin [LOGIN|CWD=répertoire [classe ...]]

Pour rendre agréable l'interaction avec l'utilisateur, ces messages ne seront affichées qu'une seule fois par session. Ainsi, le fichier *.message*, indiqué dans l'exemple ci-dessous, s'affiche à chaque passage dans un répertoire si le fichier est présent dans ce dernier, et ceci uniquement lors du premier passage.

message /welcome.msg login
message .message cwd=*

Voici un exemple de message *welcome.msg* :

Bienvenue %U (%u) !

Vous êtes connecté sur %L depuis %R.
Il y a actuellement %N utilisateur(s) de ce service.

Note : Aucun abus ne sera toléré.

En cas de problème, contactez %E

- Pour compléter le message de bienvenue, on peut ajouter derrière *cwd* un certain nombre de paramètres explicités ci-après. La valeur ' * ' les représente tous. Tous ces paramètres peuvent figurer dans le message de bienvenue.

%T	Heure locale (format *Thu Nov 15 17:12:42 1990*)
%F	Espace libre (exprimé en octets) sur la partition courante
%C	Répertoire courant
%E	**c.** Adresse de courriel de l'administrateur spécifié dans le fichier */etc/ftpaccess*
%R	Nom d'hôte du client
%L	Nom d'hôte du serveur
%u	Identification de l'utilisateur (identd, RFC 931)
%U	Nom d'utilisateur donné lors de la connexion
%M	Nombre d'utilisateurs maximum dans cette classe
%N	Nombre d'utilisateurs en ce moment dans cette classe

Il existe des paramètres spécifiques aux ratios.

%xu	nombre d'octets déposés

%xd	nombre d'octets téléchargés
%xR	ratio de téléchargement
%xc	crédit d'octets
%xT	délai limite
%xE	temps écoulé depuis le début de session (en minutes)
%xL	crédit temps
%xU	limite de dépôt
%xD	limite de téléchargement

- La directive **readme** permet de prévenir l'utilisateur qu'il existe un fichier particulier qui relate les modifications apportées au site. L'utilisateur est ainsi invité à le télécharger pour le consulter. Pour rendre agréable l'interaction avec l'utilisateur, ces messages ne seront affichées qu'une seule fois par session.

- La seule directive du même thème, non commentée ici, est **hostname**.

2.6.3. Traces (ou historiques)

- Cette directive offre la possibilité de tracer respectivement les commandes passées par les clients de type réels, ainsi que tous les transferts de fichiers des types *real* et *anonymous*, entrant et sortant, dans des fichiers journaux.

```
log commands real
log transfers real,anonymous inbound,outbound
```

Il en existe d'autres pour :
  - tracer les violations de sécurité effectuées par les clients dont le type apparaît dans la liste,
  - rediriger les traces de transfert vers le processus **syslogd**.

```
log security liste de types d'utilisateurs
log syslog
log syslog+xferlog
```

Par défaut, celles-ci ne sont écrites que dans le fichier **xferlog**. La dernière ligne redirige les traces de transfert à la fois vers *syslog* et *xferlog*.

2.6.4. Commandes diverses et utilitaires

- La directive **alias** permet de définir des raccourcis dénommant des répertoires particuliers. Dans l'exemple qui suit, l'utilisateur *nom_stagiaire* peut lancer la commande *cd nom_stagiaire* pour se déplacer sur le répertoire */Incoming/nom_stage/nom_stagiaire*.

```
alias nom_stagiaire  /Incoming/nom_stage/nom_stagiaire
```

Il est préférable d'indiquer l'existence de tels raccourcis dans l'un des messages affichés en début de session.

- Nous pouvons autoriser (ou interdire) un certain nombre d'opérations sur le site comme :
  - la compression de données (compress et gzip),
  - l'archivage (tar).

```
compress   yes      all
tar        yes      all
gzip       yes      all
```

Pour cela, le premier paramètre [yes|no] fixe l'autorisation ou l'interdiction, le deuxième précise la classe d'utilisateurs concernée par la directive.

- La directive **shutdown** avertit le client d'un arrêt du serveur. Lorsque le fichier désigné est présent, les clients sont prévenus d'une prochaine interruption du service et les nouveaux arrivant interdits d'accès.

```
shutdown /etc/shutmsg
```

La commande *ftpshut* permet de créer le fichier suivant comme il permet d'arrêter le service avec la commande suivante :

```
ftpshut now " on ferme "
```

Le fichier en question doit être structuré de façon particulière. Son contenu est une ligne du type où l'on désigne la date et l'heure précise d'interruption du service (les mois, heures et minutes sont respectivement notées de 00 à 11, 23 et 59).

```
année mois jour heure minutes interdiction déconnexion texte
```

L'heure d'interdiction de connexion est indiquée en utilisant le format HHMM. Cette mesure est relative à l'heure de fin de service (idem pour les déconnexions).

```
2001 00 20 10 00 0030 0015
Interruption de service à %s.
```

Dans l'exemple ci-dessus, le fichier ci-dessus déclare une interruption de service pour le 20 janvier 2001 à 10 heures du matin. De plus, les accès seront bloqués 30 minutes auparavant et les connexions automatiquement coupées 15 minutes auparavant.

Pour relancer le service FTP, il faut détruire ce fichier. La commande **ftprestart** fournit ce moyen automatique de redémarrer le serveur lorsque nous sommes prêts à réactiver ce dernier. Il détruit le message d'arrêt créé par la commande **ftpshut** (aussi bien le message qui concerne l'ensemble du service que celui affectant le seul service public ou tel site virtuel).

- Les autres directives du même thème sont : cdpath, daemonaddress, virtual, defaultserver, passive address, passive ports, passv-allow, port-allow, lslong, lsshort, lsplain, mailserver, incmail, mailfrom.

2.6.5. Ratios de transfert

- Il existe trois directives **ul-dl-rate**, **dl-free** et **dl-free-dir**.

2.6.6. Permissions

- Nous pouvons autoriser (ou interdire) un certain nombre d'opérations sur le site comme :
  - la modification des permissions posées sur les fichiers (chmod),
  - la suppression des fichiers (delete),
  - la modification ou l'écrasement des fichiers (overwrite),
  - le renommage des fichiers (rename),
  - le changement de masque des droits d'accès (umask).

```
chmod       no      guest,anonymous
delete      no      guest,anonymous
overwrite   no      guest,anonymous
rename      no      guest,anonymous
umask       no      guest, anonymous
```

Pour cela, le premier paramètre [yes|no] fixe l'autorisation ou l'interdiction, le deuxième précise le type d'utilisateurs concernée par la directive.

- La directive **deny -email** permet d'invalider certains mot de passe utilisés lors d'une connexion anonyme même s'ils sont conformes au contrôle réalisé. Ceux-ci sont automatiquement générés par des navigateurs mal configurés. Voici un exemple :

```
deny -email mozilla@
deny -email ie50user@
```

- La directive **passwd-check** définit le niveau de contrôle réalisé sur les mots de passe fournis par les clients anonymes.

```
passwd -check none|trivial|rfc822 enforce|warn
```

L'option **none** permet d'utiliser n'importe quelle chaîne de caractères, l'option **trivial** ne contrôle que la présence du caractère @, l'option **rfc822** assure que le mot de passe est une adresse électronique conforme au RFC 822.
Le test de conformité réalisée n'est que syntaxique.
Si le niveau de contrôle est **warn** et que le mot de passe n'est pas conforme, l'identification est acceptée mais un message sera affiché. Si le niveau de contrôle est **enforce**, alors un message d'erreur sera renvoyé et l'identification rejetée.

- La directive **path-filter** permet de contrôler quels sont les noms de fichiers autorisés lors d'un dépôt. La règle s'applique aux utilisateurs appartenant à l'un des types indiqués. En cas de non conformité, le message est affiché.

```
path-filter anonymous /usr/local/etc/pathmsg ^[-A-Za-z0-9_\.]*$ ^\. ^-
```

L'exemple permet aux utilisateurs de type *anonymous* de déposer des fichiers dont le nom contient n'importe quel suite de caractères alphabétiques majuscules ou minuscules, chiffres, point, soulignement (_) et tiret. Les seules exceptions concernent les fichiers dont le nom commence par un point ou un tiret.
Le fichier **pathmsg** contient un message indiquant l'erreur :

```
Le nom de fichier utilisé est plutôt douteux...
```

- Les autres directives du même thème sont : throughput, anonymous-root, guest-root, deny-uid, deny-gid, allow-uid, allow-gid, restricted-uid, restricted-gid, unrestricted-uid, unrestricted-gid, site-exec-max-lines, dns refuse_mismatch, dns refuse_no_reverse, dns resolveroptions.

3. Serveur Proftpd

3.1. Sites de références

Le site de référence du logiciel serveur *Proftpd* a pour adresses URL : ***www.proftpd.net*** ou ***www.proftpd.org***.

Pour récupérer l'ensemble des fichiers du serveur, il existe deux procédés :

- Celui de télécharger les sources à l'adresse ***ftp://ftp.proftpd.net/distrib/source/proftpd-1.2.7.tar.gz***,
- Celui de copier les paquetages des cédérom de distribution Linux :
  - proftpd-1.2.5-3mdk.i586.rpm (Mandrake 9.0)
  - proftpd-anonymous-1.2.5-3mdk.i586.rpm (nécessaire pour les sites publics)

Pour nous aider dans la configuration, il existe d'autres sites de référence comme :

- ***http://proftpd.crihan.fr/***.

Il existe en local une rubrique dans le **man** du système d'exploitation. Il existe aussi un répertoire complet sous ***/usr/share/doc****/proftpd*-version de fichiers d'aide dont deux pages web en anglais :

- *Configuration.html* pour la liste des paramètres de configuration,
- *FAQ.html* comme documentation complète de mise en œuvre du serveur.

3.2. Plate-formes et versions

La dernière version courante est la 1.2.7, la 1.2.8 (janvier 2003) étant en version test.

Les systèmes d'exploitation supportées sont :

AIX	BSD/OS	SunOS
DG/UX	Digital Unix	FreeBSD
HP/UX	IRIX	Linux for IBM S/390, zSeries
Linux	Mac OS X	NetBSD
OpenBSD	SCO	Solaris

3.3. Arborescence et complément

Dans les versions récentes de ce serveur, le processus démon **xinetd** remplace le processus *inetd*.

Le programme compilé du serveur se trouve sous */usr/sbin*, le script de fonctionnement du serveur sous */var/run*.
Sous LINUX, le serveur installe le répertoire pub du site de publication sous **/var/ftp**.

3.4. Paramètres globaux

En standard, la configuration est localisée dans un fichier unique : **/etc/proftpd.conf**.

3.4.1. Structure du fichier

Ce fichier se divise en plusieurs parties, qui ne sont pas toutes nécessaires (vitales). Nous avons donc plusieurs contextes de configuration :

server config, <Global>, <Anonymous>, <VirtualHost>, <Limit>, <Directory>

Nous pouvons donc rencontrer, comme sous LINUX, un fichier *proftpd.conf* contenant seulement le contexte *server config*. Le contexte *server config* n'est pas entre « <> » car il est implicite (donc inutile de le mentionner).
Les options présentes à l'intérieur des contextes de configuration sont appelées des **directives**!
Tous les contextes de configuration autres que *server config* sont forcément inclus dans ce premier contexte.
Par ailleurs, nous pouvons rencontrer un contexte comme <*Directory*> inclus dans un *sous contexte* comme <*Anonymous*>. C'est un principe de configuration imbriquée.

Dans le cas des distributions LINUX, le fichier **proftpd.conf** est le fichier central qui peut, selon les besoins, appeler des fichiers secondaires comme /etc/**proftpd-anonymous.conf** (paquetage proftpd-anonymous). Une directive d'inclusion dans le premier permet d'intégrer le second.

<*Global*> est un contexte de configuration qui peut être utilisé à l'intérieur du contexte *server config* ou du contexte <*VirtualHost*>. Tout ce qui va être défini dans <*Global*> va être appliqué à l'ensemble du contexte de configuration dans laquelle il se trouve.

La syntaxe du fichier est simple et claire : une configuration par ligne débutant par le nom du paramètre (ou directive), suivi à une tabulation près de la valeur attribuée au dit paramètre.
La syntaxe se veut semblable à celle du serveur Web Apache.
Le caractère # sert toujours à désactiver une ligne de configuration en la mettant en commentaire.

3.4.2. Service standard

Pour un service minimum autorisant les utilisateurs réels ayant un compte UNIX, le fichier doit posséder les lignes suivantes :

```
1    ServerName                  " Proftpd Server "
2    ServerType                   standalone
```

3	**Port**	**21**

1. Le fichier commence dès la première ligne par préciser le nom du serveur qui apparaîtra dans la bannière d'accueil du client.

```
Connected to vmlinux90.
220 ProFTPD 1.2.Oprel0 Server (Proftpd Server) [vmlinux90.esat.terre.def]
```

2. Nous fixons à cet endroit le mode de fonctionnement et d'écoute du serveur : autonome (cas général et donc choix par défaut) ou subordonné au processus réseau *inetd*. La valeur *inetd* peut aussi désigner le nouveau démon réseau *xinetd*.

3. Il s'agit du numéro de port de service TCP par défaut. Le changement de numéro de port intéresse les administrateurs qui publient plusieurs sites (virtualhosts) dont des sites privés.

Les lignes qui suivent sont d'importance secondaire.

4	**User**	**nobody**
	Group	**nogroup**
5	**Umask**	**022**
6	**MaxInstances**	**30**
7	**LsDefaultOptions**	**"-l"**

4. Les deux lignes concernent le compte d'utilisateur interne au fonctionnement du serveur, ainsi que son groupe d'appartenance.

5. La valeur de cette option correspond au standard UNIX en matière de masque des droits des fichiers à leur création. La valeur préconisée ci-dessus empêche les droits d'écriture sur les répertoires et fichiers du serveur au niveau des clients (droits des niveaux groupe et/ou autres utilisateurs). Une autre valeur possible est *022 022*, où le premier argument de la directive concerne les fichiers et le second les répertoires.

6. Afin de se prémunir d'attaques DOS, l'administrateur du serveur a intérêt à indiquer le nombre maximum.

7. Nous activons ici l'option par défaut pour le listage des fichiers (équivalent à la commande *dir* du système d'exploitation DOS).

```
<Directory /*>
AllowOverwrite      on
</Directory>
```

Le premier contexte <Directory> porte sur le répertoire racine du site « /* ». Le répertoire « /* » représente la valeur données au contexte, valeur glissée à l'intérieur de cette première balise. Ces contextes formant bloc commencent par une balise d'ouverture et se terminent par une balise de fermeture. Toutes les instructions se trouvant entre ces deux balises ne s'appliquent qu'au contexte en question.

Ici, la directive *AllowOverwrite on* autorise l'écrasement des fichiers existants (ou le remplacement). Cette directive est inutile si l'écriture est interdite (<LIMIT WRITE>).

Une fois que nous avons redéfini certains paramètres, il est nécessaire de faire relire au serveur le fichier de configuration modifié. Pour cela, il faut arrêter le service (./proftpd stop), puis le redémarrer (./proftpd start).

Cela permet aussi de tester notre nouvelle configuration. Si des erreurs sont présentes, il ne relance pas le service et nous prévient par des messages d'avertissement.

```
./proftpd start
        Lancement du serveur FTP (proftpd) :        -fatal : unknown configuration
        directive '(Anonymous)' on line 46 of '/etc/proftpd.conf'
```

Ou

```
./proftpd start
        Lancement du serveur FTP (proftpd) :    -fatal : Allow : directive not allowed in
        <Anonymous> context
```

Une fois que le démarrage du service est réussi, nous pouvons tester la connexion au serveur en utilisant un compte UNIX (nom d'utilisateur, mot de passe).

```
D:\> ftp vmlinux90.esat.terre.def
Connected to vmlinux90.esat.terre.def.
220 ProFTPD 1.2.Oprel0 Server (Proftpd Server) [vmlinux90.esat.terre.def]
Name (vmlinux90): moi
331 Password required for moi.
Password:
230 user moi logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 Port command successful.
150 Opening ASCII mode data connection for file list.
drwxr-x---      3 moi          users          4096 Dec 19 14:18 irclogs
226-Transfer complete.
226 Quotas off
ftp>
```

Il se peut que l'identification échoue pour une raison ou pour une autre. Des messages adaptés sont alors retournés au client.

```
530 Login incorrect.
Echec de l'identification.
```

3.5. Site public

3.5.1. Téléchargement

Un site FTP public permet de mettre des archives et des fichiers à la disposition de tous. En réalité, "tous" n'existe pas dans un système de type UNIX. De ce fait, lorsqu'un site FTP est

ouvert à tous, nous utilisons le compte d'utilisateur *anonymous* associé au vrai compte UNIX **ftp**.
Ceci n'est guère visible si le client utilise un navigateur Web pour accéder aux serveurs FTP publics. En fait, le navigateur utilise le compte *anonymous* de manière transparente afin d'éviter au client la procédure de connexion. En résultat, il affiche la liste des fichiers mis à sa disposition.

```
1     <Anonymous ~ftp>
        AllowAll
2       User            ftp
        Group           ftp
        UserAlias       anonymous ftp
3       MaxClients      10
4       DisplayLogin            welcome.msg
        DisplayFirstChdir       .message
5       RequireValidShell       off
      </Anonymous>
```

Le bloc <Anonymous> enferme tous les paramètres propres à la mise en œuvre d'un site public :

1. La valeur donnée au contexte *~ftp*, désigne le répertoire de connexion de l'utilisateur *ftp* (*/home/ftp*). Il sert de répertoire d'accueil ou de racine au site.

2. Le fichier déclare dans un premier temps le seul compte du système UNIX **ftp** autorisé au profit des clients FTP. Il indique dans un deuxième temps son groupe système d'appartenance. Enfin, il précise l'alias **anonymous** reconnu conventionnellement.

3. La directive **MaxClients** limite le nombre de clients ou de connexions possibles à 10.

4. Les directives **DisplayLogin** et **DisplayFirstChdir** ont pour but d'afficher respectivement le message de bienvenue à la connexion, ainsi que celui affiché à chaque déplacement dans l'arborescence du serveur.

5. La directive **RequireValidShell** désactive ici tous les interpréteurs de commandes cités dans le fichier */etc/shells*, de façon à empêcher les clients de passer n'importe quelle commande UNIX sous l'invite *ftp*.
   Elle peut être activée dans le cas où l'utilisateur autorisé ne bénéficie que d'un *shell* limité ou dans le cas où l'utilisateur n'est autre que l'administrateur système pouvant user de tous ses droits.

Une fois que la configuration du site public est terminée, nous pouvons tester la connexion au serveur.

```
Utilisateur (vmlinux90.esat.terre.def :<none>) : anonymous
331 Anonymous login OK, send your complete email address as your password.
Mot de passe :
230 Anonymous access granted, restrictions apply.
ftp> ls
200 PORT command successful.
```

```
150 Opening ASCII mode data connection for file list.
Pub
226 – Transfer complete.
226 Quotas off.
ftp : 5 octets reçus dans 0,02 secondes 0,31 Ko/sec.
```

3.5.2. Dépose

Les sites FTP publics ne limitent pas les clients au seul téléchargement. Il arrive qu'ils proposent aux clients de déposer leurs contributions à la communauté du Net en termes d'images, de graticiels ou autre.
Une configuration supplémentaire est nécessaire pour que les utilisateurs *ftp* et *anonymous* puissent déposer (ou uploader) dans le répertoire communément désigné *Incoming* ou *uploads*. La mise en place de ce type de fonctionnalité ne présente pas de difficultés particulières.
Il suffit en effet de spécifier des permissions particulières pour le répertoire en question à l'aide des contextes de type <Limit>.
A l'intérieur du bloc *Anonymous* précédent, nous définissons dans un premier temps, une interdiction d'écriture pour l'ensemble des fichiers se trouvant sous */home/ftp* (valeur * à l'intérieur de la balise) :

```
<Directory *>
        <Limit WRITE>
        DenyAll
        </Limit>
</Directory>
```

Un nouveau bloc de balises <Directory> fait suite au précédent pour localiser le répertoire désigné pour la dépose. A l'intérieur de ce dernier, nous posons les permissions spécifiques au service de dépose :

- interdire la lecture du contenu du répertoire (<Limit READ>), afin que les utilisateurs anonymes ne puissent pas se servir de ce répertoire comme lieu d'échange de fichiers ;
- autoriser le stockage des fichiers (<Limit STOR>).

```
<Directory uploads/*>
                    <Limit READ>
                        DenyAll
                    </Limit>
                    <Limit STOR>
                        AllowAll
                   </Limit>
</Directory>
```

La valeur *uploads/** autorise la dépose sous le répertoire */home/ftp/uploads*.

Une autre variante consiste à :

- autoriser les déplacements dans tous les répertoires de dépose à l'aide de la balise <Limit CWD>, accompagnée de la directive *AllowAll*,.

- interdire successivement la lecture, la suppression des répertoires et des fichiers, la création de répertoires à l'aide de la balise <Limit READ RMD DELE MKD>, accompagnée de la directive *DenyAll*.

```
<Directory /home/ftp/uploads>
        AllowAll
  <Limit STOR CWD>
        AllowAll
  </Limit>
  <Limit READ RMD DELE MKD>
        DenyAll
  </Limit>
</Directory>
```

En pratique, il faut dans l'ordre :

- créer le répertoire *Incoming* ou *uploads*,
  - Il est possible de le créer autant sous le répertoire *pub* que sous la racine du serveur public.
  - En cas de dysfonctionnement, l'administrateur est obligé de vérifier que le nom du groupe propriétaire du répertoire est ***ftp***. Dans le cas négatif, il lance la commande : *chgrp ftp* uploads.
- modifier le fichier de configuration comme indiqué ci-dessus,
- tester la simple dépose de fichier sur le serveur,
  - Si le message *Permission denied* est retourné au client, l'administrateur doit vérifier que les droits en écriture soient posés sur le répertoire en question au niveau du groupe (et/ou autre). Dans le cas négatif, il lance la commande : *chmod go+w* uploads.
- vérifier que les permissions posées soient bien appliquées : DELETE, MKDIR, etc…

```
230 Anonymous access granted, restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd incoming
250 CWD command successful.
ftp> put ecole.pdf
local: ecole.pdf remote: ecole.pdf
200 PORT command successful.
150 Opening BINARY mode data connection for ecole.pdf.
226 Transfer complets. 263461 bytes sent in 0.321 secs (8e+02 Kbytes/sec)
ftp> get ecole.pdf
local : ecole.pdf remote: ecole.pdf
200 PORT command successful.
550 ecole.pdf: Permission non accordée
ftp> bye
221 Goodbye.
```

Remarque :
L'interdiction de lecture étant posée, le client ne peut plus télécharger (get) le fichier déposé.

Complément :

La directive **AllowStoreRestart** autorise les clients à reprendre les déposes en cas d'interruption de transfert. Elle ne concerne pas les reprises de téléchargements depuis le serveur.

3.5.3. Service FTP anonyme seul

Pour limiter le service FTP au seul accès anonyme, nous pouvons ajouter deux limitations. Tout d'abord, au niveau du contexte « server config », nous interdisons l'accès à tous :

```
<Limit LOGIN>
        DenyAll
</Limit>
```

Ainsi, en principe, plus personne ne peut se connecter par FTP sur le serveur.
A l'intérieur du contexte *Anonymous*, nous plaçons une directive inverse :

```
<Limit LOGIN>
        AllowAll
</Limit>
```

Si un utilisateur du système tente de se connecter au serveur, il sera rejeté.

```
# ftp vmlinux9.esat.terre.def
Connected to vmlinux9.esat.terre.def
220 ProFTPD 1.2.Oprel0 Server (Proftpd Server) [vmlinux9.esat.terre.def]
Name (vmlinux90:<none>): user1
331 Password required for user1.
Password:
530 Login incorrect.
```

3.5.4. Association utilisateurs – répertoires

L'une des possibilités du serveur Proftpd est de permettre un accès direct à certains répertoires en fonction du nom d'utilisateur employé pour la connexion au serveur. Nous imaginons les alias suivants pour l'utilisateur **ftp**, en plus de l'alias *anonymous* :

```
UserAlias       ftpl ftp
UserAlias       ftpi ftp
UserAlias       ftpf ftp
UserAlias       ftpv ftp
```

Nous conservons l'alias *anonymous* par souci de compatibilité avec les navigateurs Web, mais nous ajoutons quatre nouveaux alias. Ceux-ci sont destinés à être utilisés respectivement pour les accès aux archives de logos, images, fond d'écran et vidéos.

Remarque :
On regrettera ici le fait de ne pas pouvoir définir plusieurs alias sur une seule ligne. En effet, la directive *Useralias* ne prend en argument, dans l'ordre, qu'un alias suivi d'un nom d'utilisateur valide au niveau du système.

Ensuite, nous créons quatre répertoires dans */home/ftp* portant le nom des alias. Enfin, nous ajoutons dans notre fichier de configuration (à l'intérieur de la portée du serveur *Anonymous*) la ligne suivante :

```
UserDirRoot      on
```

Dès le redémarrage du serveur, un utilisateur employant le compte alias *anonymous* ou le compte réel *ftp* sera connecté comme avant dans */home/ftp*. En revanche, grâce à cette dernière directive, s'il utilise par exemple le nouvel alias *ftp1*, il arrivera sur un serveur dérouté vers */home/ftp/ftpl*.

3.5.5. Utilisateurs avec privilèges

Même dans le cas d'un serveur offrant un accès public, il peut être bienvenu qu'un certain nombre de personnes soient autorisées à manipuler certains fichiers et répertoires sur le serveur, au minimum l'administrateur de site.
Il n'est pas nécessaire pour autant qu'il s'agisse d'utilisateurs reconnus par le système UNIX à distance ou même localement. De plus, même si l'utilisateur possède effectivement un compte système (ou réel) sur la station hébergeant le serveur, il n'est pas nécessaire que le répertoire personnel coïncide avec celui auquel il accède par FTP.

Pour clarifier tout cela, nous prenons l'exemple d'un serveur FTP public destiné à partager les travaux d'un groupe de travail. Les membres du groupe en question sont *monchef* et *moi*. Ces deux utilisateurs offrent publiquement des fichiers récupérables par n'importe quel invité, mais seuls nos deux membres ont le droit de modifier les données en question à distance.

Comme solution, nous choisissons de créer à la racine du serveur FTP un répertoire pour chaque membre du groupe. Nous avons donc dans le répertoire */home/ftp* :

```
drwxr-xr-x   5   ftp       nogroup     4096   jan 22  10:05  ./
drwxrwsr-x   5   root      root        4096   jan 21  19:05  ../
drwxr-xr-x   2   monchef   monchef     4096   jan 23  10:05  monchef/
drwxr-xr-x   2   ftp       nogroup     4096   jan 22  19:12  pub/
drwxr-xr-x   2   ftp       nogroup     4096   jan 22  19:12  incoming/
drwxr-xr-x   2   moi       users       4096   jan 23  09:59  moi/
-rw-r--r--   1   root      root         166   fév 24   2001  welcome.msg
```

Nous notons au passage que les utilisateurs ont toutes les permissions sur leurs répertoires alors que les *autres* (membres du groupe ou autres groupes) peuvent uniquement y entrer et lire le contenu.

Au niveau du fichier de configuration, le contexte <Anonymous ~ftp> contient pour le moment, par exemple, les lignes suivantes :

```
<Anonymous ~ftp>
        User                ftp
        Group               nogroup
        UserAlias           anonymous    ftp
        RequireValidShell   off
        MaxClients          10
        DisplayLogin        welcome.msg
```

```
        DisplayFirstChdir    message
        <Directory *>
          <Limit WRITE>
              DenyAll
          </Limit>
        </Directory>
        <Directory incoming>
           <Limit READ WRITE>
              DenyAll
           </Limit>
           <Limit STOR>
              AllowAll
           </Limit>
        </Directory>
</Anonymous>
```

En dehors du contexte <Anonymous>, nous ajoutons :

```
RequireValidShell     off
DefaultRoot           /home/ftp
DefaultChdir          /home/ftp
```

La directive **DefaultChdir**, qui est importante, permet de désactiver le fait que, par défaut, un utilisateur légitime du système arrive directement dans son répertoire personnel. Ici, nous le déroutons directement vers la racine du serveur FTP public.
La directive **DefaultRoot** l'empêche de remonter dans l'arborescence.
Les permissions données sur les répertoires des utilisateurs interdisent aux utilisateurs invités d'écrire dans les répertoires en question. Nous n'avons donc pas besoin d'employer la balise *Limit* pour configurer ce point.
Les utilisateurs *monchef* et *moi* peuvent allègrement modifier, ajouter et supprimer des fichiers dans les répertoires */home/ftp/monchef* et */home/ftp/moi*, tout en autorisant les visiteurs à récupérer les fichiers qui s'y trouvent.

Variante

Nous imaginons à présent que l'administrateur du serveur FTP ait pour tâche d'éviter que des données légalement douteuses soient diffusées par les utilisateurs *monchef* et *moi* via leur répertoire. Il doit nécessairement posséder la permission de supprimer des fichiers dans ces répertoires.
Le problème qui se pose est le suivant : il s'agit de permettre à *monchef* et *moi* d'écrire dans leur répertoire respectif tout en permettant à un utilisateur que nous appellerons *master* de faire de même dans les mêmes répertoires. Bien sûr, il ne faut pas que *monchef* puisse écrire dans le répertoire de moi et inversement.
Nous ne pouvons plus nous baser sur les permissions Unix car, même en utilisant un groupe commun aux trois utilisateurs, nous n'arriverions pas au résultat escompté. Le système de permissions du serveur *Proftpd* permet de franchir cet écueil.

Une première étape consiste à rendre le serveur propriétaire des ressources de ces répertoires.

```
drwxr-xr-x   5    ftp      nogroup    4096   jan 22  10:05   ./
drwxrwsr-x   5    root     root       4096   jan 21  19:05   ../
```

```
drwxrwxrwx 2    ftp      nogroup   4096  jan 23 10:05  monchef/
drwxr-xr-x  2   ftp      nogroup   4096  jan 22  19:12  pub/
drwxr-xr-x  2   ftp      nogroup   4096  jan 22  19:12  incoming/
drwxrwxrwx 2    ftp      nogroup   4096  jan 23  09:59  moi/
-rw-r--r--  1   root     root       166  fév 24   2001  welcome.msg
```

Nous changeons également les permissions afin de permettre à tous les utilisateurs d'écrire, de lire et de parcourir ces répertoires. Sans modification du fichier de configuration, n'importe qui pourrait maintenant faire ce qu'il désire dans ces répertoires. Nous nous empressons donc d'ajouter dans le fichier de configuration (en dehors du contexte *Anonymous*) les lignes suivantes :

```
<Directory /home/ftp/moi>
        <Limit WRITE>
                DenyUser !master,!moi
        </Limit>
</Directory>

<Directory /home/ftp/denis>
        <Limit WRITE>
                DenyUser !master,!denis
        </Limit>
</Directory>
```

Nous utilisons une balise *Limit* pour chacun des répertoires concernés et faisons usage d'une directive nouvelle : **DenyUser**. Celle-ci permet, à l'instar de *DenyAll*, d'interdire l'action spécifiée par la balise Limit, à la différence qu'elle prend en argument un ou plusieurs noms d'utilisateurs. Le « ! » est le signe de la négation. Notre restriction définit donc une interdiction en écriture dans les répertoires pour tous les utilisateurs excepté l'administrateur du serveur (master) et le gestionnaire du répertoire en question.
Nous aurions également pu utiliser les lignes suivantes :

```
<Directory /home/ftp/moi>
    <Limit WRITE>
        DenyAll
    </Limit>
</Directory>
<Directory /home/ftp/moi>
     <Limit WRITE>
        AllowUser master,moi
    </Limit>
</Directory>
```

La première syntaxe peut être préférée car elle est beaucoup plus concise et elle facilite la maintenance.
Le résultat est le suivant.

- Les utilisateurs invités peuvent lire et récupérer le contenu des répertoires *moi* et *monchef*.
- Les utilisateurs *moi* et *monchef* peuvent écrire dans leur répertoire respectif uniquement.
- L'utilisateur *master* possède un droit d'écriture dans les répertoires *moi* et *monchef*.

3.6. Serveur Virtuel

Le logiciel *Proftpd*, à l'image d'autres serveurs, propose la configuration de serveur virtuel ou hôte virtuel (virtualhost).

Cette fonctionnalité peut être mise en œuvre, quelle que soit le nombre d'identifiants réseau (adresses IP) de la station hébergeant le serveur.

3.6.1. Serveur à une seule adresse IP

Ce serveur peut bénéficier de plusieurs alias à son nom de domaine entièrement qualifié (FQDN ou « nom de station ».« nom de domaine DNS ») dans le serveur DNS de rattachement.
Ainsi, l'utilisateur peut se connecter sur tel site FTP public ou privé selon le nom de site d'accès. Chaque site peut posséder en conséquence des paramètres de gestion différents.
Le procédé est identique à celui des serveurs Web Apache.

Nous avons pris l'exemple d'un site virtuel privé dont l'administrateur est le seul client. La configuration attendue dans le fichier **/etc/proftpd.conf** est du type :

```
<VirtualHost ftp.monsite.com>
        ServerName              "FTP Perso"
        <Limit LOGIN>
                DenyAll
        </Limit>
        <Anonymous ~webmaster>
                User            webmaster
                Group           ftpwww
                AnonRequirePassword    on
                <Limit LOGIN>
                        AllowAll
                </Limit>
                HideUser        root
                HideGroup       root
        </Anonymous>
</VirtualHost>
```

Le répertoire racine est déterminé par l'attribut de la balise *Anonymous* : *~webmaster* donc le répertoire de connexion */home/webmaster*. L'utilisateur FTP *webmaster* fait partie du groupe FTP *ftpwww*.
Comme il s'agit en fait du site FTP de l'administrateur du serveur, celui-ci prend les droits de l'administrateur système *root* (directive **HideUser** et **HideGroup**).
Les utilisateurs qui tenteraient une connexion sur le serveur par le biais de l'adresse IP de la station seraient rejetés.

La directive **AnonRequirePassword**, une fois activée, impose au client de fournir le mot de passe associé au compte d'utilisateur système créé au préalable.

En pratique, il faut dans l'ordre :

1. Créer un groupe d'utilisateurs UNIX *ftpwww* grâce à la commande :

groupadd *ftpwww*

2. Créer un utilisateur *webmaster* en le rattachant au groupe précédent grâce à la commande :

useradd –g *<GID du groupe ftpwww>* -d *//home/webmaster* –s *//bin/bash webmaster*

3. Donner un mot de passe à ce nouveau compte et communiquer l'ensemble aux clients potentiels

passwd webmaster

4. Mettre en œuvre la configuration décrite ci-dessus
5. Relancer le serveur
   - Dans le cadre des essais, le serveur, au lancement, vérifie que le nom du site est bien défini dans le fichier de résolution locale (*/etc/hosts*) en l'absence de serveur DNS correctement paramétré.
6. Côté station cliente, paramétrer le fichier *hosts* avec la ligne suivante :

160.192.xxx.yyy ftp.monsite.com

Lors des essais, le serveur prend en compte les modifications dès la réponse d'identification du serveur :

220 ProFTPD 1.2.Oprel0 Server (FTP Perso) [ftp.monsite.com]

3.6.2. Serveur à deux adresses IP

Nous prenons par exemple :

- le site *ftp.serveur_proftpd_local.com* associé à l'adresse IP 192.168.10.10,
- le site *ftp.serveur_proftpd_internet.com* associé à l'adresse IP 159.159.159.159.

Ces deux adresses IP concernent la même machine.

Il est possible de spécifier un numéro de port différent ou le même numéro (ou de ne pas le spécifier si nous employons celui du contexte *server config*).
Il est possible de faire fonctionner plusieurs serveurs virtuels (<VirtualHost>) sur la même adresse IP mais avec un numéro de port différent. Mais attention, cette possibilité est incompatible avec la configuration **ServerType inetd**.
De plus, le client doit indiquer le numéro de port dans sa procédure de connexion.

```
<VirtualHost ftp.serveur_proftpd.com>
        ServerName      "Mon serveur FTP virtuel"
        Port            46000
        Maxclients      3
        MaxClientsPerHost 1
        DefaultRoot     ~
        AccessGrantMsg "Bienvenue %u sur le serveur virtuel de la DMSI"
        <Limit LOGIN>
                AllowUser toto
```

```
            DenyAll
        </Limit>
</VirtualHost>
```

Le site virtuel ci-dessus est donc accessible au seul utilisateur *toto* s'il connaît le nom du site et le numéro de port 46000.

La directive **MaxClientsPerHost** limite le nombre de clients pour le même compte d'utilisateur. Si nous utilisons la directive *MaxClients*, il faut forcément que la valeur de la directive *MaxClientsPerHost* y soit strictement inférieur.

3.7. Administration

3.7.1. Gestion des utilisateurs

a. Utilisateurs réels

Comme stratégie, nous pouvons choisir d' **utiliser le fichier /etc/ftpusers** pour définir tous les utilisateurs qui n'ont pas accès au service FTP. Pour indiquer au serveur *Proftpd* d'utiliser ce fichier, nous activons la directive **UseFtpUsers** au niveau des paramètres globaux du serveur. En fait, il le fait par défaut.

Pour **interdire le service public** (accès anonyme), il suffit de rajouter *anonymous* dans le fichier */etc/ftpusers*.

b. Utilisateurs FTP

Comme stratégie, nous pouvons aussi choisir de créer des utilisateurs spécifiques au service FTP.
Ces utilisateurs ne doivent pas, par mesure de sécurité, bénéficier d'un interpréteur de commandes (shell) leur permettant de se connecter en telnet, etc...
Pour cela, nous attribuons à l'utilisateur en question le shell */bin/false* (interpréteur bidon), paramètre modifié dans le fichier */etc/passwd*.
Pour que ce shell soit *valide*, il faut mentionner son existence dans le fichier */etc/shells*, si ce n'est pas déjà fait.
En effet, le serveur *Proftpd* par défaut, n'accepte pas la connexion si le shell de l'utilisateur n'est pas indiqué dans */etc/shells*.
De plus, nous pouvons spécifier lors dans le fichier de configuration *proftpd.conf* si nous souhaitons cette vérification. Cela se fait par le biais de la directive : **RequireValidShell**.

3.7.2. Directives complémentaires

Voici quelques directives complémentaires que nous pouvons insérer au niveau des paramètres globaux du serveur. Elles intéressent l'administrateur éclairé.

1	**AccessGrantMsg**	**"Bienvenue %u sur le serveur DMSI"**
2	**DeferWelcome**	**off**
3	**MultilineRFC2228**	**on**
4	**ShowSymlinks**	**on**
5	**AllowOverwrite**	**on**

```
6   TimeoutNoTransfer 600
    TimeoutStalled    600
    TimeoutIdle       1200
```

1. La directive **AccessGrantMsg** personnalise le message d'accueil (%u est un paramètre qui récupère le nom d'utilisateur qui se connecte).

2. La directive **DeferWelcome** active (off) ou désactive (on) le fait de retirer le contenu de la directive *ServerName* de la bannière. Par exemple, le serveur n'affichera plus que les messages suivants.

```
Connected to vmlinux90.
220 ProFTPD 1.2.Oprel0 Server ready.
```

3. La directive **MultilineRFC2228** permet de compenser les problèmes d'affichage des lignes issus du RFC 959, le RFC 2228 assurant une meilleure compatibilité avec les navigateurs Web.

4. La directive **ShowSymlinks** permet au visiteur de voir que tel fichier ou tel répertoire est un lien symbolique ou non.

5. **AllowOverwrite** est une directive dangereuse car elle permet à un visiteur d'écraser un fichier (si ses droits le permettent). Par défaut, un client FTP ne peut pas écraser un fichier existant, cette option permet donc de passer outre.

6. Les trois directives font partie des mesures de sécurité à prendre au niveau d'un serveur FTP. Ils déterminent le temps limite pour un client connecté qui n'engage pas de transferts, pour un transfert de données "bloqué" *(stalled)* et pour un client qui ne fait rien. Les valeurs sont données en nombre de secondes.
   Sans fixer de valeurs, le serveur applique de lui-même ces directives avec des valeurs par défaut que seule la lecture du fichier de configuration modifie. Après le lancement d'une nouvelle commande, nous obtenons alors un message du type :

```
421 No Transfer Timeout <300 seconds> : closing control connection.
Connexion fermée par l'hôte distant.
```

7. La directive **DefaultServer** peut servir dans le cas de la mise en œuvre des hôtes ou serveurs virtuels à la manière du serveur Web Apache. Si le seul serveur à activer est le serveur virtuel configuré, alors nous désactiverons (off) le serveur principal (ou par défaut). Il est cependant conseillé de la laisser active (on).

```
7   DefaultServer     on
8   DefaultRoot       ~
9   ServerIdent       "ProFTP DMSI Server Ready"

10  PersistentPasswd  off
11  MaxLoginAttempts 3
```

8. La directive **DefaultRoot** complète la sécurité d'accès aux fichiers système de la station qui héberge le serveur. Elle peut être aussi bien insérée au niveau de la

configuration globale qu'au niveau des sites privés. Elle limite l'utilisateur connecté à la racine par défaut du site (*home directory* ou répertoire de connexion pour les utilisateurs réels).
En résultat, l'utilisateur ne peut plus remonter dans le répertoire de niveau supérieur. Bien qu'il n'obtienne aucun message d'erreur ou d'avertissement, il ne pourra accéder qu'aux répertoires de niveaux inférieurs.
Un serveur public limite automatiquement le parcours des répertoires à la racine du serveur.

9. La directive **ServerIdent** modifie le contenu du premier message affiché lors d'une tentative de connexion sur notre serveur. Le client obtient ce message même si sa tentative échoue. Si nous mettons cette option à *off*, le client verra le message suivant :

   "[hostname] FTP server ready.".

   La valeur du paramètre *hostname* sera souvent le nom complet de la station (*localhost.localdomain* par exemple).
   Il est recommandé de ne pas annoncer le type de logiciel utilisé comme serveur FTP. En effet, nous ne sommes jamais à l'abri d'un éventuel bogue de sécurité. De ce fait, en ajoutant cette ligne, nous ne fournissons pas d'informations à d'éventuelles personnes malveillantes.

10. La directive **PersistentPasswd** activée (on) permettrait au serveur *Proftpd* de valider lui même les mots de passe des comptes UNIX en utilisant les fichiers de configuration */etc/passwd* et */etc/shadow*. Dans le doute, nous laissons cette directive désactivée.

11. Par souci de sécurité, la directive **MaxLoginAttempts** permet de limiter le nombre de tentatives de connexions.

```
12  <Limit MKD RNFR RNTO DELE RMD STOR CHMOD SITE_CHMOD SITE
    XCUP WRITE XRMD XPWD>
        DenyAll
    </Limit>

13  AllowForeignAdress on

14  <Limit LOGIN>
        Allow 172.16.18.5 192.168.10.
        Deny all
    </Limit>

15  RateReadBPS 14000

16  AllowGroup  ftpwww
    AllowUser   webmaster
    DenyUser    pirate
```

12. Le contexte **LIMIT** a déjà été décrit plus haut, mais se voit ici complété par l'ensemble des paramètres. Il délimite les permissions posées sur les sites ou sous-sites.

    RNFR : (rename from) empêche de renommer les fichiers.
    RNTO : (rename to) est le paramètre contraire au précédent.

13. La directive **AllowForeignAdress** autorise littéralement le client à oeuvrer depuis un autre ordinateur que le sien. Ceci revient à autoriser une personne A à transférer des fichiers entre un serveur B et notre serveur C. Cette directive doit être activée avec précaution car elle autorise le client à utiliser des logiciels de **type FXP**, ouvrant le serveur à des attaques.

14. Il est possible de filtrer les adresses IP fixes d'un réseau local en mettant en oeuvre un contexte **<LIMIT LOGIN>** ainsi que la directive **Allow**. Il est déconseillé de mettre une filtre sur un nom de domaine pour des raisons évidentes de sécurité même si cela peut paraître une solution de facilité. L'exemple ci-dessus filtre une adresse IP donnée (172.16.18.5) ainsi qu'une classe d'adresses IP (192.168.10.x).

15. La directive **RateReadBBS** est un exemple d'options qui permet de limiter la bande passante en lecture.

16. De la même manière que la directive **AllowGroup** autorise des groupes d'utilisateurs, les directives **AllowUser** et **AllowAll** autorisent respectivement un utilisateur FTP donné ou tous les utilisateurs définis.
    A l'inverse, les directives **Deny…** interdisent l'accès à tous, un groupe donné, un utilisateur donné.

3.7.3. Interroger le serveur en local

La consultation du *man* du système nous apprend que l'on peut interroger le serveur en lançant la commande *proftpd* avec certaines options :

- *proftpd –v* pour obtenir la version du serveur,
- *proftpd –l* pour découvrir la liste des modules compilés,

```
Compiled-in modules :
mod….c
```

- *proftpd –t* pour tester la configuration avant de lancer le serveur,

```
Checking syntax of configuration file.
Syntax check complete.
```

- *proftpd –p* pour rendre persistant les mots de passe,
- *proftpd –c* pour indiquer un autre fichier de configuration que proftpd.conf,
- *proftpd –d* suivi d'un chiffre entre 0 et 5 pour niveler le débogage du fonctionnement du serveur.

3.7.4. Surveiller le serveur

Le serveur *Proftpd*, comme bien d'autres serveurs Internet, permet l'archivage des journaux d'activité. Ces journaux représentent une partie importante de la gestion des serveurs car ils permettent de surveiller l'activité et la popularité du serveur. Ils sont surtout la principale source de renseignements en cas de problème de sécurité ou d'attaque externe.
Par défaut, en l'absence de configuration spécifique, les informations envoyées dans le fichier *syslog*, se limitent à ceci :

```
proftpd [1003]              : svrftplx1 - ProFTPD 1.2.Oprel0 standalone mode STARTUP
svrftplx1 proftpd [988]     : svrftplx1 (vmlinux[192.168.0.10])- FTP session closed.
svrftplx1 proftpd[1004]     : svrftplx1 (vmlinux[192.168.0.10])- FTP session closed.
proftpd[11321]              : svrftplx1 (vmlinux[192.168.0.10]) - USER nini (Login failed):
Can't find user.
svrftplx1 proftpd[10131     : svrftplx1 (vmlinux[192.168.0.10]) FTP session closed.
```

En bref, nous apprenons peu de choses, si ce n'est qu'un utilisateur provenant de l'hôte *vmlinux* s'est déconnecté, qu'un utilisateur *nini* s'est vu refusé car il n'existe pas de compte sous ce nom sur le serveur.

Le serveur *Proftpd* permet de définir des fichiers journaux personnalisables grâce à la directive **ExtendedLog**.

```
ExtendedLog /var/log/proftpd-access.log  WRITE,READ
ExtendedLog /var/log/proftpd-auth.log    AUTH
ExtendedLog /var/log/proftpd-all.log     ALL
```

Le premier argument de la directive indique le nom et la localisation du nouveau fichier journal. Le second argument, séparé par une espace, est un mot clef spécifiant le type d'informations à stocker dans le fichier.
Nous définissons ici trois fichiers. Le premier archivera des informations sur les activités de lecture et écriture sur le serveur FTP. Le second archivera tout ce qui concerne l'authentification des utilisateurs. Enfin, le dernier contiendra absolument toutes les informations découlant d'une connexion à notre serveur.

D'autres mots clefs peuvent être utilisés :

- INFO pour toutes les opérations de demandes d'informations (commandes PWD, SYST, etc.).
- DIRS pour toutes les opérations concernant les répertoires (listage, création, suppression, etc.).
- MISC pour toutes les opérations inclassifiables des autres catégories.

Les informations ainsi envoyées dans les journaux sont formatées automatiquement par le serveur *Proftpd,* mais il est possible de personnaliser le formatage à l'aide d'une directive spécifique : **LogFormat**.

Cette directive prend en paramètre une chaîne composée de méta-séquences (débutant par le symbole %) et de chaînes de caractères fixes. Les méta-séquences permettent de spécifier des informations spécifiques à placer pour composer une ligne du journal d'activité du serveur FTP :

%A Le nom d'utilisateur anonyme (son adresse de courriel donnée en tant que mot de passe) ou UNKNOWN s'il s'agit d'un utilisateur légitime.

%b Le nombre d'octets envoyés pour une requête.

%f Le nom du fichier récupéré (RETR) ou envoyé (STOR). Il s'agit du chemin absolu.

%F Le nom du fichier récupéré (RETR) ou envoyé (STOR) tel que le client le voit.

%h Le nom de l'hôte distant (client).

%a L'adresse IP de l'hôte distant.

%l Le nom d'utilisateur distant ou UNKNOWN si la requête *ident* échoue.

%m La commande reçue de la part du client (RETR, LIST, STOR, etc.).

%p Le port local utilisé par le serveur.

%v Le nom d'hôte local du serveur.

%P Le PID du serveur.

%r La ligne de commande complète envoyée par le client.

%T Le temps de réception/émission pour un fichier en secondes.
%s Le code de réponse numérique (200, 404,etc.).

%u L'ID utilisateur d'un visiteur légitime et authentifié.

Il nous est possible dès à présent de composer les formats de journaux en associant toutes ces méta-séquences :

```
LogFormat fmtaccess "%h %l %u %t \"%r\" %s %b"
```

Nous notons que la chaîne complète est entre guillemets. Si nous souhaitons faire apparaître le symbole *guillemet* dans les fichiers journaux, il nous faudra le faire précéder ici d'un anti-slash. Il en va de même avec le % qui devra être écrit dans la définition de format des fichiers journaux.
Le second argument de la directive *LogFormat* est un nom de format arbitrairement choisi par nos soins. Nous pouvons ensuite le réutiliser ainsi :

```
ExtendedLog     /var/log/proftpd-access.log     WRITE,READ fmtaccess
```

Le fichier *proftpd-access.log* respectera le nouveau format personnalisé *fmtaccess*.

3.8. Conclusion

Il reste beaucoup de choses à dire sur *proftpd* mais ce document n'a pas la prétention d'être une documentation officielle. Les informations données ici devraient être suffisantes pour la

mise en oeuvre d'un serveur FTP viable et fonctionnel. Pour plus d'informations, il est conseillé de se reporter directement au site officiel.

Chapitre 3 : Serveur Windows Serv-U

1. Présentation

Serv-U est un serveur FTP simple mais puissant. Il présente des options de sécurité comme la restriction de l'accès au dossier, le blocage d'adresses IP, la surveillance en temps réel et l'enregistrement (fichiers journaux) des actions des utilisateurs. Il offre aussi la reprise des transferts de fichiers interrompus.

Le document a été écrit sur la base de la dernière version du moment, la 4.04.

1.1. Caractéristiques

En se basant sur la publicité de la société *RhinoSoft.com (**www.rhinosoft.com**)*, auteur de ce logiciel, voici les principaux traits de ce serveur :

- Mise en œuvre de **Secure-FTP** à travers **SSL/TLS**.
- Mesures de sécurité comprenant les mots de passe, les permissions lire/écrire/ajouter /modifier par répertoire ou fichier pour chaque utilisateur (incluant Anonymous), et des restrictions d'accès basés sur les adresses IP.
- Multiple **serveurs FTP 'virtuel'** peuvent être installés avec une seule instance de Serv-U. Ces serveurs peuvent être **administrés à distance**.
- Support des **mots de passe à un jeton S/KEY**.
- Peut fonctionner comme un service système natif sous Windows NT, 2000 et 95/98/ME/ XP.
- Support de **comptes temporaires** qui sont automatiquement supprimés après expiration.
- Support des **ratios UL/DL**, des limitations en **quota de disques**, de la limitation de la **bande passante réseau**, et des **mesures automatiques contre les procédés de hackers** que sont les techniques d'anti-délai et hammering.
- Possibilité de mise en oeuvre des chemins UNC.
- Possibilité de mise en oeuvre de chemins virtuels : répertoires ou lecteurs peuvent être liés à n'importe quel répertoire racine de site privé d'utilisateur.
- Utilisateurs peuvent être insérés dans des groupes pour une administration aisée.
- Une complète implémentation des standards FTP avec les RFC959, RFC1123, RFC1760, RFC2228, RFC2246, RFC2289, RFC2389 et le document de travail sur Secure-FTP.
- Active les délais d'inactivité ou de connexion, de sorte que les connexions sont automatiquement closes dans plusieurs cas de figure.
- Facile à configurer et à maintenir grâce à un **programme séparé pour l'administrateur**.
- Présente une architecture ouverte qui permet la surveillance, les modifications et les extensions possible à travers l'ajout de DLL externes.
- Archive toutes les transactions dans des **fichiers journaux**.

1.2. Editions

Le logiciel Serv-U se présente en trois éditions :

- L' "Edition Personnelle", libre d'utilisation, fournit toutes les fonctionnalités nécessaires à un usage personnel.

- L' "Edition Standard" a été promue comme la solution de partages de fichiers pour la plupart des hommes d'affaires.
- L' "Edition Professionnelle" est faite pour des sites FTP pour grands comptes qui nécessite la mise en œuvre de nombreux serveurs virtuels FTP ainsi qu'un trafic de fichiers important en volume.

Pour une présentation générale plus complète des éditions Serv-U, il faut se rendre sur le site **www.Serv-U.com**.

2. Installation

2.1. Conseils préalables

2.1.1. Licence utilisateur

Une fois que le choix est fait, il faut consulter la liste située à l'adresse ***shop.watsoft.net***.
Dès que nous avons décidé quelle licence nous convient, nous imprimons notre bon de commande et nous le faisons parvenir accompagné du règlement.

2.1.2. Clé de la licence

Après avoir lancé l'utilitaire d'administration de SERV U, nous ouvrons l'arborescence *Local Server* et nous cliquons sur *Licence*. Nous saisissons nos informations de licence, et nous validons avec le bouton *Enter Key*.

2.2. Installation proprement dite

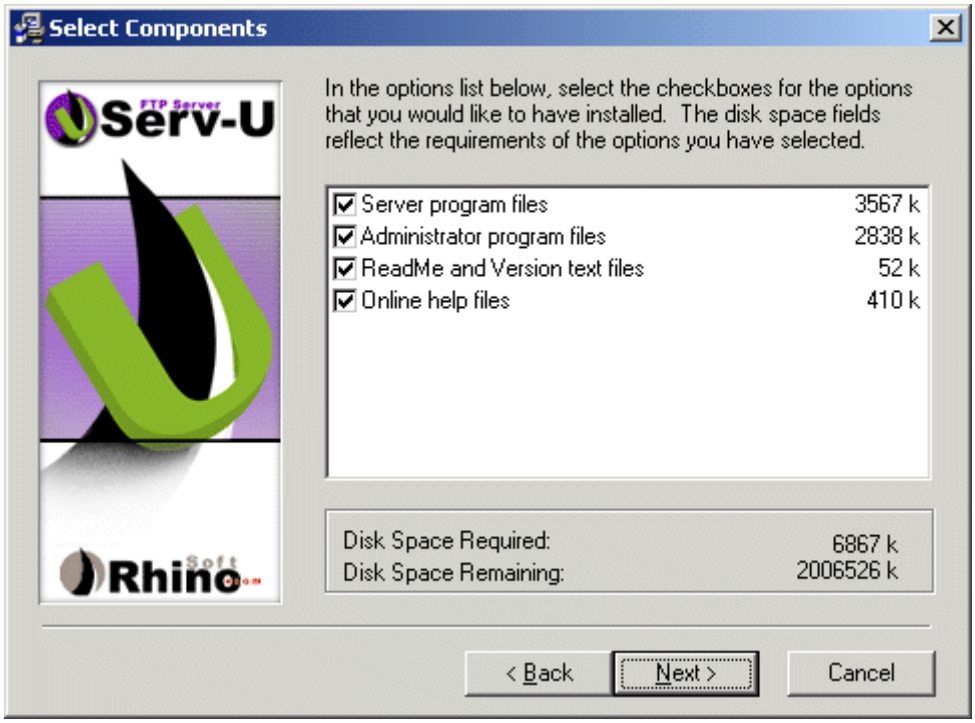
Il suffit de cliquer sur le fichier *susetup.exe* pour lancer l'installation.

Le premier écran ci-après conseille de sauvegarder l'installation d'une version antérieure.

Le deuxième écran fait de la promotion pour l'utilisation de FTP-secure, une version sécurisée du service FTP mettant en œuvre la technique des certificats et le protocole SSL (Secure Socket Layer). Ce serveur propose l'emploi natif de ce service.

Le troisième demande à ce que soit accepté le contrat de licence avant de poursuivre l'installation. Il suffit de cocher la case proposée pour activer le bouton *Suivant*.

Le quatrième écran demande à localiser le lieu d'installation, par défaut sous *Program Files*.



Le cinquième rappelle les différentes composantes du serveur que nous nous apprêtons à installer :

- Les fichiers propres au serveur lui-même,
- Les fichiers d'administration du serveur,
- Les fichiers d'aide en ligne.

L'écran suivant nous demande dans quel groupe de programmes installer ce dernier. Puis il nous confirme le début d'installation.

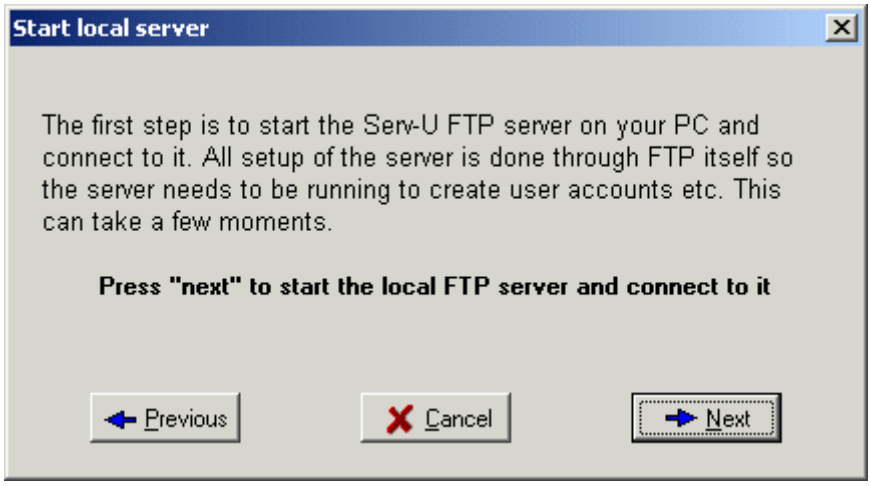
L'avant-dernier écran nous félicite pour le choix effectué et effectue la publicité sur d'autres programmes fournis par la même société.

Le dernier écran nous propose de poser une icône du serveur sur le bureau, et de lancer l'assistant d'aide à la configuration du serveur.

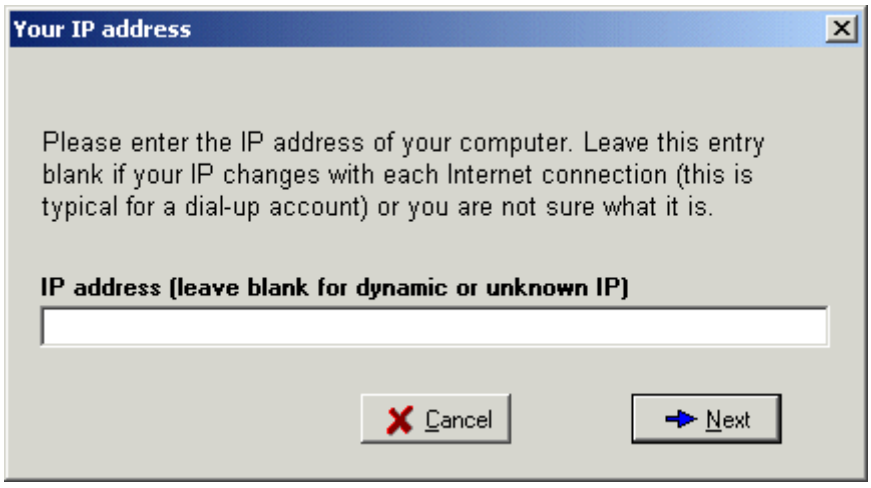
2.3. Configuration initiale

Un premier écran nous expose le but de cet assistanat avant de continuer ou d'annuler cette procédure.

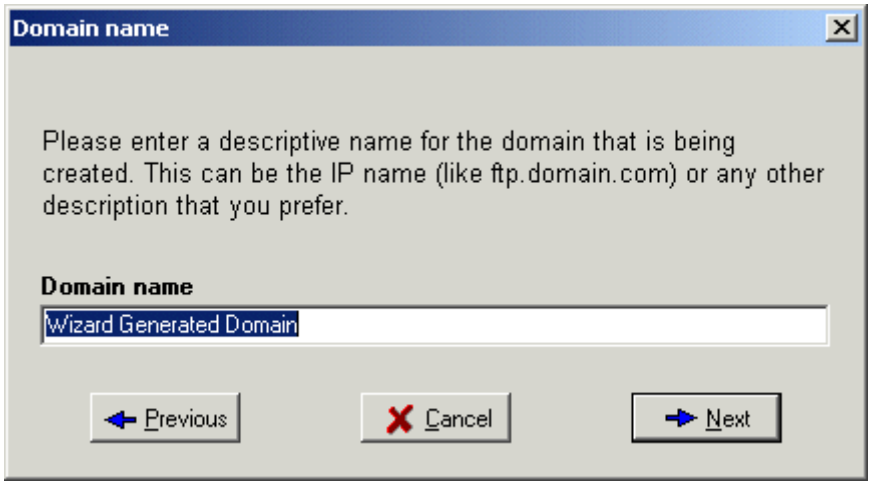
Le deuxième concerne l'affichage des icônes et leur taille. Les non-voyants doivent choisir la configuration par défaut.



Le troisième nous propose de démarrer le service pour pouvoir se connecter au serveur et donc le paramétrer.



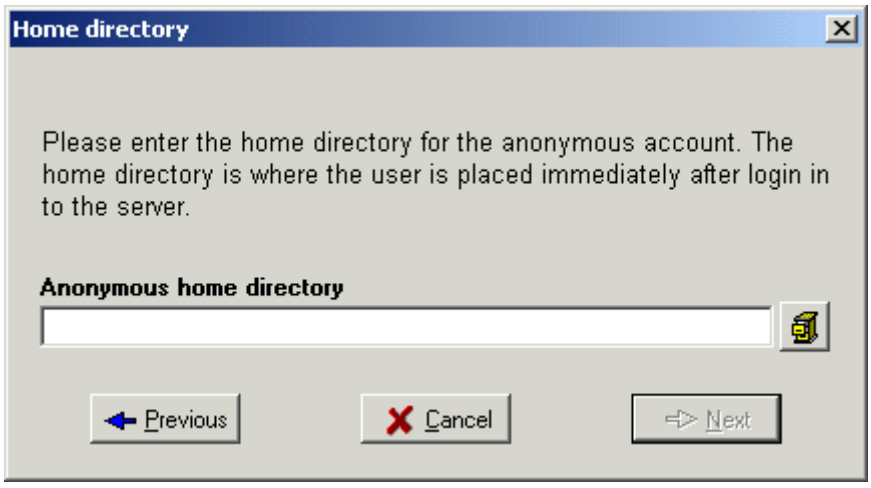
L'écran ci-dessus ainsi que l'écran suivant concerne les identifiants réseau du serveur : son adresse IP et son nom de domaine.



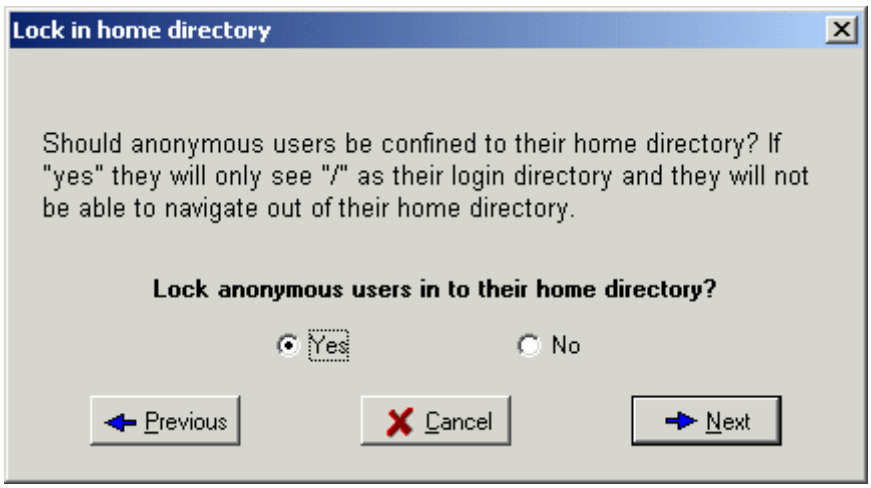
Les différents écrans suivants concernent la configuration du site public :

- La gestion des accès ;

- La localisation du répertoire d'accueil ;



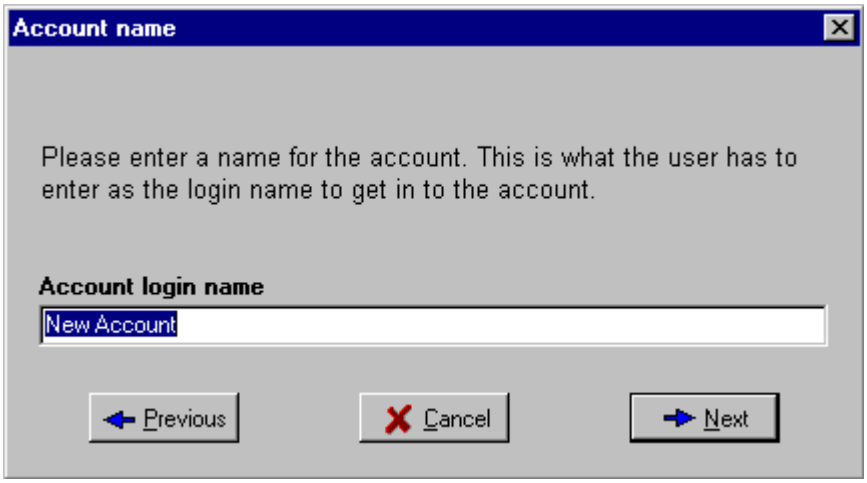
- Le blocage des utilisateurs anonyme au niveau de la racine du site en question (fortement conseillé).
    - Il ne peut qu'accéder aux répertoires de niveaux inférieurs.



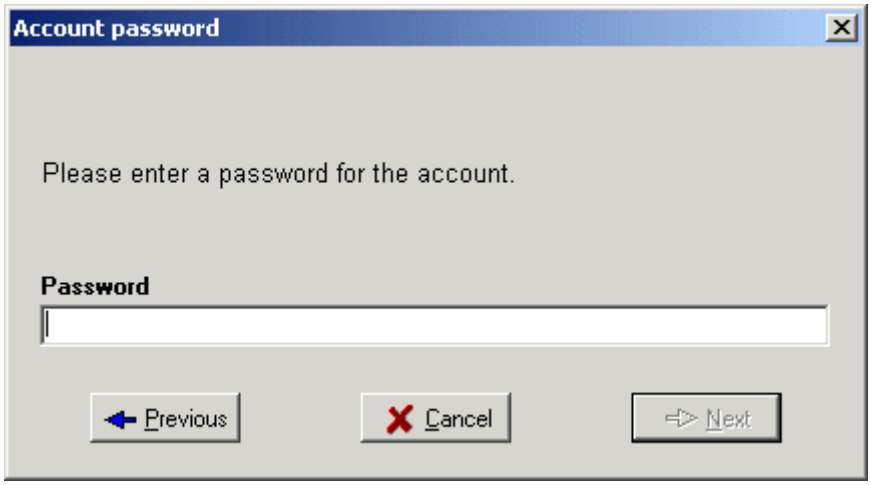
La suite de la configuration concerne les sites privés pour lequel un compte d'utilisateur est défini (**Named Account**).

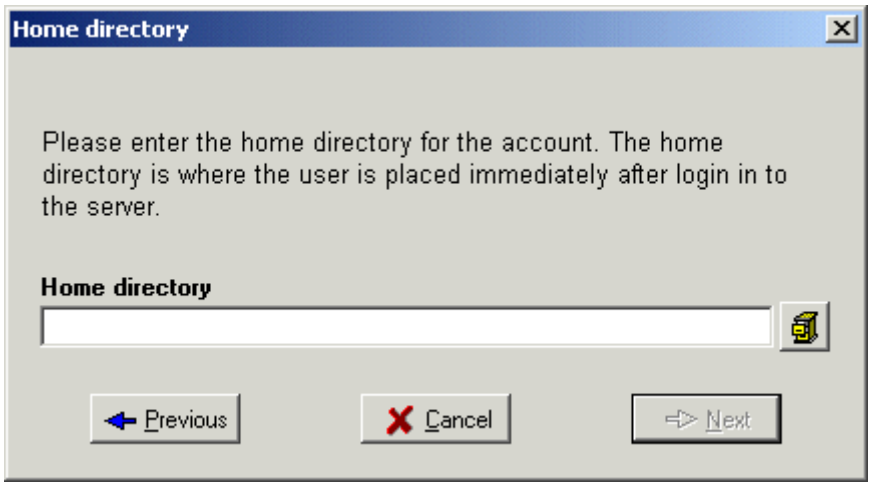
La configuration d'un tel site exige de fixer les éléments suivants :

- Le nom d'utilisateur du compte privé ;



- Le mot de passe ;
- La localisation du répertoire d'accueil.

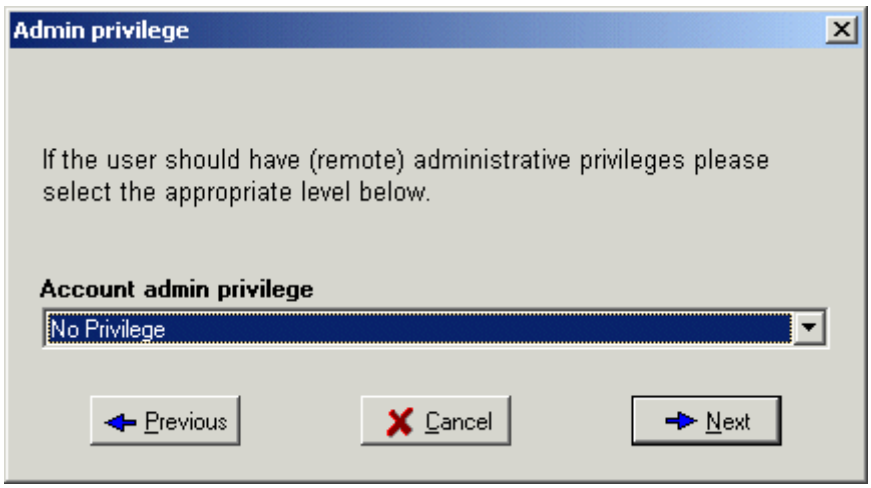




Une boite de dialogue nous demande si nous voulons confiner l'utilisateur à son répertoire de connexion, comme précédemment pour le compte *anonymous*.

Dans la boite de dialogue suivante, l'administrateur peut décider s'il s'agit enfin de décider si cet utilisateur peut disposer de certains privilèges, par exemple :

- celui de déposer des fichiers dans la structure créée pour lui,
- ou bénéficier de tous les privilèges d'administrateur pour permettre à ce dernier de pouvoir à distance déposer les ressources sur le site public, gérer le contenu des sites privés.



Une dernière boite de dialogue annonce que la configuration initiale du serveur est terminée. Un bouton *Previous* nous permet de revenir en arrière sur la configuration, un bouton *Finish* de terminer la configuration.

L'administrateur peut lancer à loisir le programme d'administration pour modifier les paramètres de tel ou tel site. Ceci fait l'objet de la partie suivante.

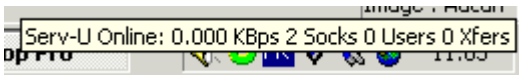
2.4. Fichiers installés

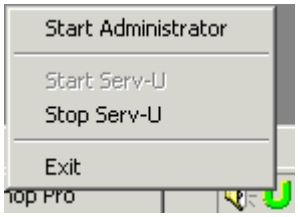
2.4.1. Bureau

L'icône du programme d'administration est déposée sur le bureau.

2.4.2. Programmes résidents

Une autre icône apparaît dans la barre d'état du système à droite.
En survolant l'icône, un message d'informations renseigne sur le trafic actuel du serveur.





L'icône change d'aspect selon les évènements :

- Le U devient vert (voir figures ci-dessus) lorsque aucune session et transfert est en cours.

- Le U est barré par un sens interdit quand le serveur est arrêté.

- Le U est simplement barré lorsque le service est en train de redémarrer, ou de s'arrêter.

- Le U devient bleu quand des transferts sont en cours.

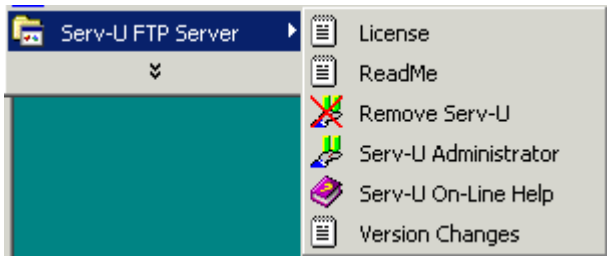
- Le U devient bleu et bordé de rouge quand le transfert est terminé mais la session non close.

En activant le menu contextuel, nous pouvons :

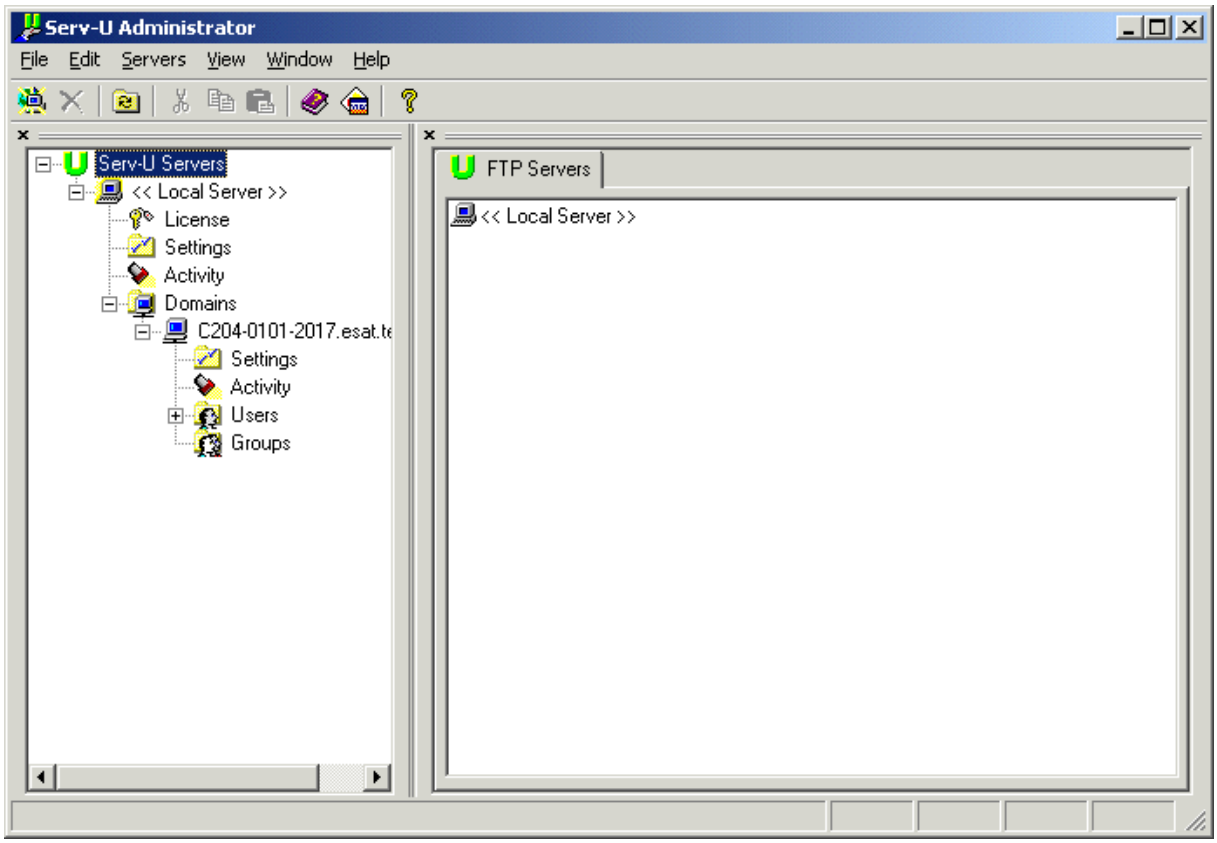
- arrêter (ou démarrer le serveur),
- accéder au programme d'administration.

2.4.3. Menu Démarrer

Le Menu Démarrer / Programmes propose un sous-menu comportant les accès suivants :



- un fichier texte sur la licence du serveur,
- un fichier texte d'informations (readme)
- le programme de désinstallation
- le programme d'administration
- l'aide en ligne (fichier *.hlp)
- un fichier texte sur les changements apportés par cette version



3. Administration

3.1. Présentation

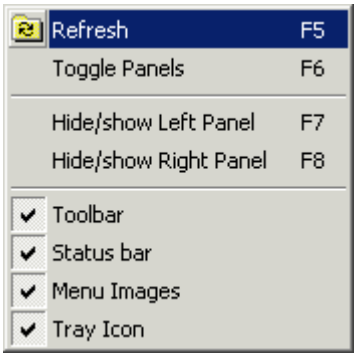
Le logiciel d'administration, dont une copie d'écran se trouve dans la page précédente, présente une barre de menus, une barre d'outils et deux volets.

3.1.1. Barre de menus

Le menu *File* donne accès à la seule option *Exit* pour quitter le programme.
Le menu *Edit* offre les fonctions traditionnelles d'édition : couper, copier et coller.

Le troisième menu s'adapte non seulement au nœud sélectionné dans le volet de recherche, mais aussi à l'onglet sélectionné dans le volet de résultat dans le cas de plusieurs onglets.
Le menu contextuel du volet de résultat reprend le menu spécifique à chaque nœud ou à chaque onglet de ce volet, menu présent aussi bien dans la barre de menus que dans la barre d'outils.
La description de ce menu sera réalisée dans chaque paragraphe qui traite de chacun des nœuds de configuration.

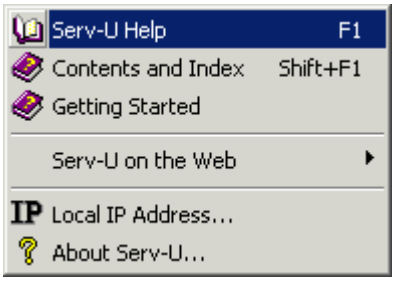
Au niveau du nœud *Serv-U Server*, le menu spécifique *Server* permet d'installer un nouveau serveur.



Le menu *View* ouvre plusieurs fonctions :

- rafraîchir l'affichage des volets (*refresh*),
- personnaliser les menus (*toggle panels*)
- cacher ou afficher l'un des deux volets (*hide/show*…),
- désactiver les barres d'outils et d'état (*tool* et *status bar*),
- désactiver les petites icônes de la barre d'outils au profit de grandes (*menu images*),
- désactiver l'icône résidente de la barre d'état (*tray icon*).

Le menu *Window* permet de consulter les messages envoyés par un utilisateur du serveur à l'administrateur.



Le menu *Help* présente d'abord trois options d'aide mettant en œuvre le même fichier d'aide.
Dans un deuxième temps, il offre un sous-menu exigeant une connexion sur le réseau Internet avant d'accéder aux différents documents proposés.
Dans un troisième temps, il permet de constater les adresses IP d'accès au serveur.
Enfin, il renseigne sur la société qui fournit le logiciel.

3.1.2. Barre d'outils

La barre d'outils s'adapte au nœud sélectionné dans le volet d'exploration, et de temps à autre à l'onglet choisi dans l'onglet de résultat.

Cependant, toutes les barres proposent en commun et dans l'ordre les options suivantes :

- supprimer un nœud ou un composant,
- rafraîchir l'information,
- couper, copier, coller une donnée,
- consulter l'aide, acheter le logiciel, obtenir des renseignements sur le produit et la société.

Nous commentons à titre d'exemple celle du niveau *Server-U*.
Si nous enlevons les options décrites ci-dessus, nous découvrons deux nouvelles icônes.

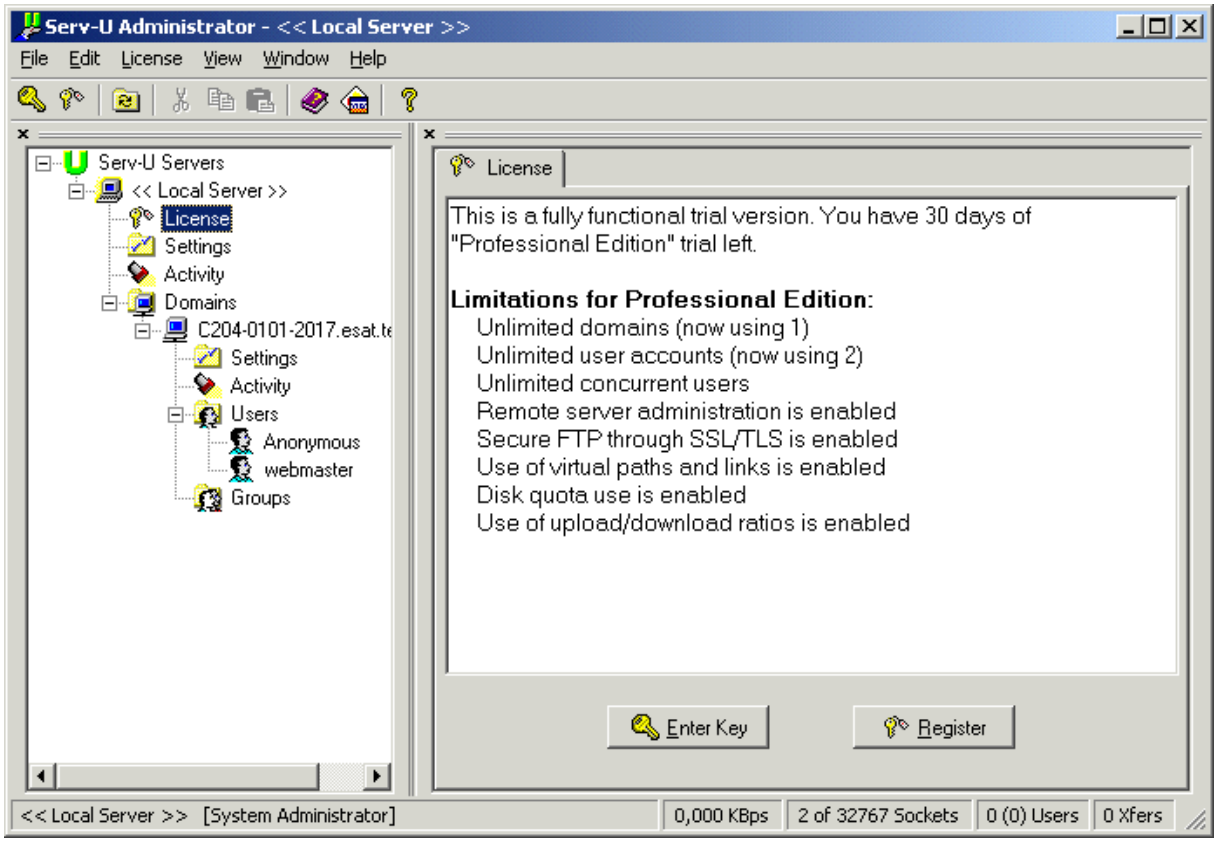
L'icône toute à gauche ne fait que reprendre celle du menu spécifique *Server*. L'icône de droite permet d'accéder à la liste des messages.

3.1.3. Volets d'exploration et de paramétrage

Le volet de gauche expose une arborescence qui débute par la racine icônisée *Serv-U Servers*, puis continue avec l'icône *Local Server*.
La présence de cette icône semble indiquer que ce logiciel peut administrer à distance les services FTP d'autres stations.

Sur la station locale, les paramètres sont regroupés en quatre catégories :

- *Licence* pour la gestion des licences,
- *Settings* pour les paramètres globaux du serveur,
- *Activity* pour surveiller l'activité des clients et filtrer les accès,
- *Domains* pour gérer les différents sites publics et privés du serveur.



Le volet de droite voit son contenu évoluer en fonction du nœud sélectionné dans l'arborescence de gauche.

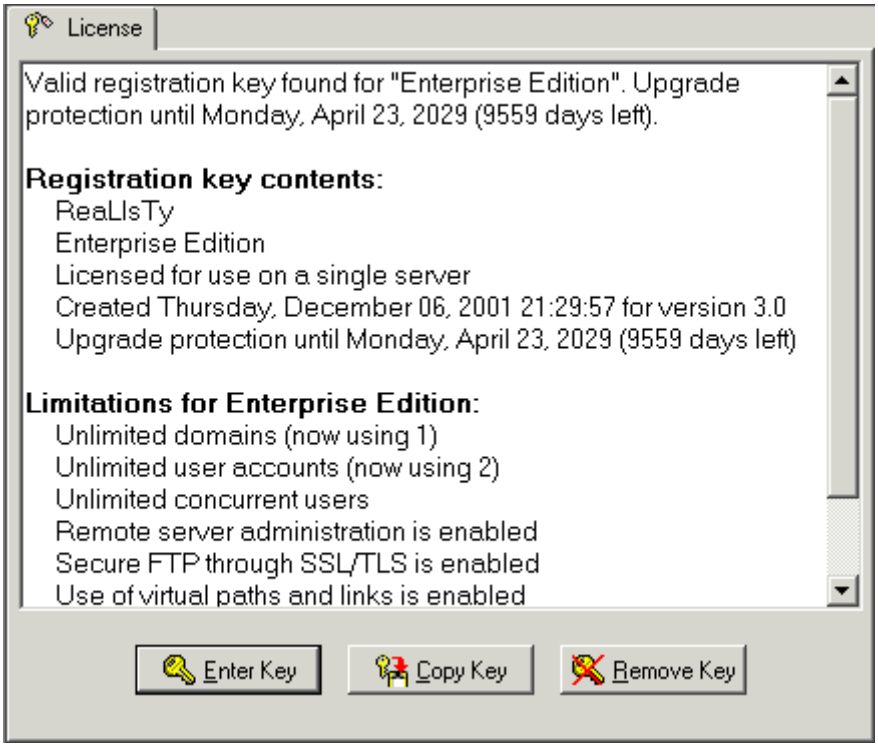
3.1.4. Barre d'état

Le contenu de la barre d'état est invariable quelle que soit le nœud sélectionné.
Les quatre zones de droite renseignent successivement sur la bande passante utilisée, le nombre de *sockets* en cours d'utilisation (ou numéros de ports), le nombre d'utilisateurs connectés et le nombre de transferts en cours.
La partie la plus à gauche, cependant, affiche un message informatif en fonction du survol de la souris.

3.2. Licence

La première chose à faire est d'enregistrer la licence du serveur.

La clé d'enregistrement se situe dans un fichier texte fourni avec le logiciel. L'utilisateur doit copier dans le Presse-papiers la clé en question, puis cliquer sur le bouton *Enter Key*. La validation du logiciel est alors visible dans la copie d'écran suivante.



Le menu *License* présente trois options permettant successivement d'entrer la clé, de la copier et de la supprimer. La copie permet de charger la clé dans le *presse-papiers* de Windows, clé que l'on peut coller dans un fichier texte sous *Bloc-notes* à titre de sauvegarde.

La barre d'outils offre les même options.

Le menu contextuel du volet de résultat reprend celui de la barre de menus. Dans la zone d'affichage du volet de résultat, il est possible de couper, copier ou coller tout morceau de texte.

3.3. Identité du serveur

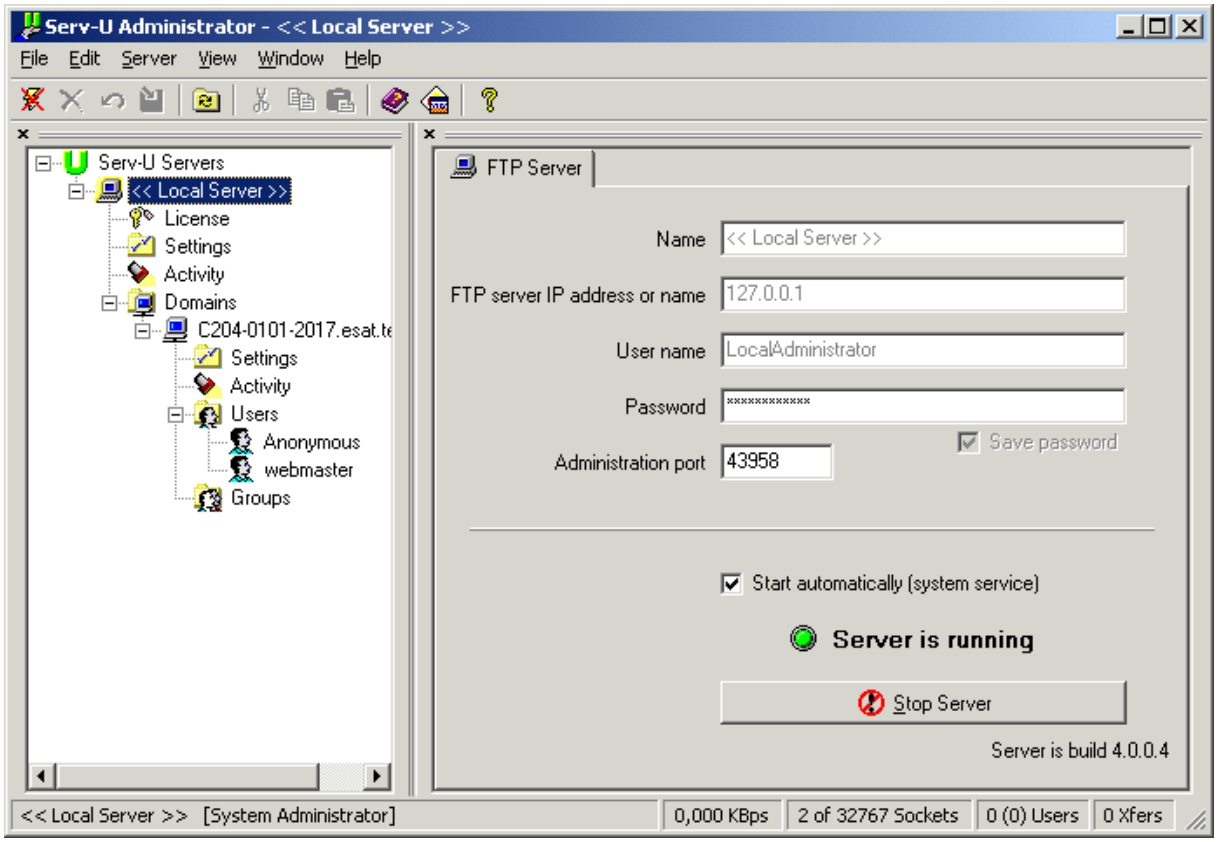
Le choix du nœud *Local Server* dans le volet de gauche fait apparaître un seul onglet dans le volet de droite où l'on précise les identifiants du serveur :

- Son nom (étiquette pour le programme d'administration à ne pas changer) ;
- Son adresse IP (adresse de boucle locale dans le cas du serveur local),
  - Un nom de domaine DNS, préalablement configuré dans le serveur du même nom, est possible en remplacement de l'adresse IP.
- Le compte d'utilisateur du serveur utile pour l'administration à distance de celui-ci ;
  - Le nom de *Local Server* impose *LocalAdministrator*. Un compte d'utilisateur reconnu par le serveur distant suffit. Cet utilisateur doit avoir quelques privilèges d'administrateur pour configurer le serveur.
- Le mot de passe correspondant, que l'on peut sauvegarder chiffré en local ;
- Le numéro de port TCP **confidentiel** pour permettre cette administration à distance.

L'administrateur peut, à l'aide d'un bouton du volet de droite, comme dans *Panneau de Configuration / Services*, arrêter et démarrer le serveur dans cet onglet.

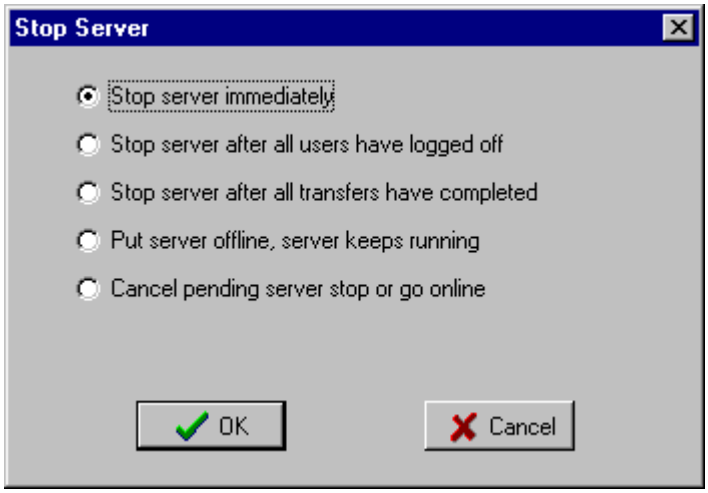
L'administrateur peut revenir sur la décision prise lors de l'installation pour gérer le serveur comme un service démarré automatiquement au lancement du système ou non (case à cocher). Il est conseillé d'arrêter le service avant de décocher la case pour ne pas provoquer de dysfonctionnement.

En bas de l'onglet à droite figure le numéro de version du serveur, utile pour le support technique.



La barre de menus ainsi que la barre d'outils offre une option *Server* pour déconnecter le dit serveur (arrêter le service).
L'action du bouton *Stop Server* lance une nouvelle fenêtre à options.

L'option *Stop server immediately* n'attend pas que les utilisateurs soit déconnectés et que les transferts en cours soient terminés. Seul le programme d'administration peut alors redémarrer le serveur.
L'option *Stop server after all users have logged off* attend les déconnexions des clients avant d'arrêter le service.
L'option *Stop server after all transfers have completed* attend la fin des transferts avant d'arrêter le service.



L'option *Put server offline, server keeps running* permet de prendre des précautions à l'égard de l'administration des serveurs distants. Cette option met hors service (offline) le serveur tout en autorisant les sessions en cours ainsi que les transferts. Seuls les nouveaux utilisateurs seront bloqués dans leur tentative de connexion.
La dernière option *Cancel pending server stop or go online* permet l'action inverse à l'option précédente *(online server*) ou l'annulation d'un arrêt en cours suite au choix d'une des trois premières options.

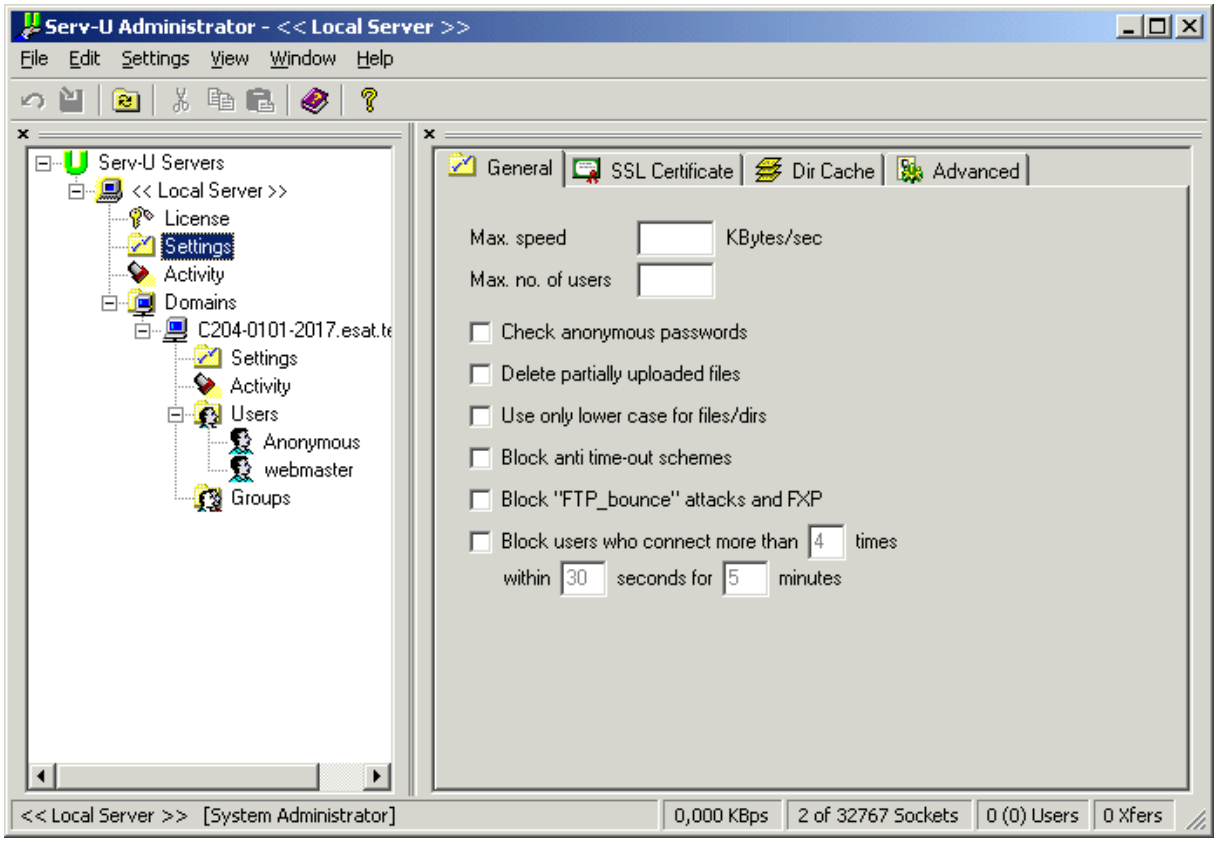
*3.4. Paramètres globaux (*Settings*)*

Pour passer en revue les paramètres généraux, il faut cliquer sur *Settings*. Dans le volet de droite apparaissent quatre onglets :

- Onglet *General* pour déclarer les principaux paramètres,
- Onglet *SSL Certificate* pour mettre en œuvre des procédés de chiffrement en ce qui concerne les accès et les transferts,
- Onglet *Dir Cache* pour gérer le cache du serveur,
- Onglet *Advanced* pour configurer la sécurité Internet.

Le principal onglet reste l'onglet *General*.

3.4.1. Onglet *General*

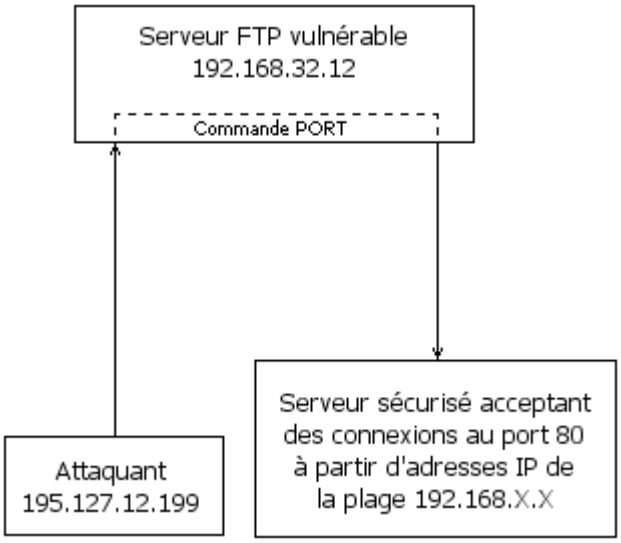


Commentons les différents paramètres.

Max speed	Indique la vitesse maximum disponible pour les transferts (dépose et téléchargement).
Max no. of users	Fixe le nombre total d'utilisateurs pouvant se connecter.
Delete partially uploaded files	Lorsqu'un fichier déposé n'est pas complet, le serveur peut le supprimer.
Use only lower case ...	Les dossiers et fichiers sont affichés en minuscules.
Block anti time-out schemes	Empêche le client d'utiliser la fonction ***anti time-out*** de son logiciel client ftp. Avec cette fonction, le client peut rester connecté tout le temps désiré, au détriment des autres clients. Ceci a pour conséquence d'une part de ne faciliter pas la gestion du service FTP par l'administrateur, et d'autre part de rendre le serveur vulnérable aux attaques de *hackers*.
Block "FTP_bounce" and FXP	Empêche le client d'attaquer indirectement le serveur par procédé ***bounce*** (1) et d'utiliser les fonctionnalités ***fxp*** d'un logiciel client (le procédé *fxp* a été présenté sommairement dans le premier chapitre du document).
Block users who ...	Empêche le client de se connecter plusieurs fois de suite sur le serveur à des intervalles de temps très réduits : procédé d'attaque dit ***hammering***.

(1) ***Bounce***
Le procédé *FTP Bounce* signifie *Rebond FTP*. Il est basé sur une utilisation de la commande PORT du protocole FTP lorsque le serveur

FTP est en mode actif. En effet, cette commande permet de se connecter à n'importe quel autre serveur distant, et à un port donné. Il est possible que la sécurité du serveur cible soit compromise dans le cas où ce dernier filtre les adresses IP entrantes. En effet, l'adresse IP que le serveur cible verra sera l'adresse IP du serveur intermédiaire, et non l'adresse IP de l'attaquant.
Ce petit schéma explique la technique utilisée :



Conséquences :

- Vol d'identité
- Permet d'accéder à des données confidentielles

Comment s'en protéger ?

- Dû au fait que l'attaque est compatible avec le protocole (cf RFC), la politique de sécurité peut-être variable selon les implémentations.
- Nous suggérons de supprimer la commande PORT.

(2) ***Hammering***
Ce procédé consiste à se connecter plusieurs fois de suite sur un serveur à des intervalles de temps très réduits.

3.4.2. Onglet *SSL Certificate*

Cet onglet est valide dans le cas de l'utilisation d'un serveur de certificats entre le serveur FTP et son client.

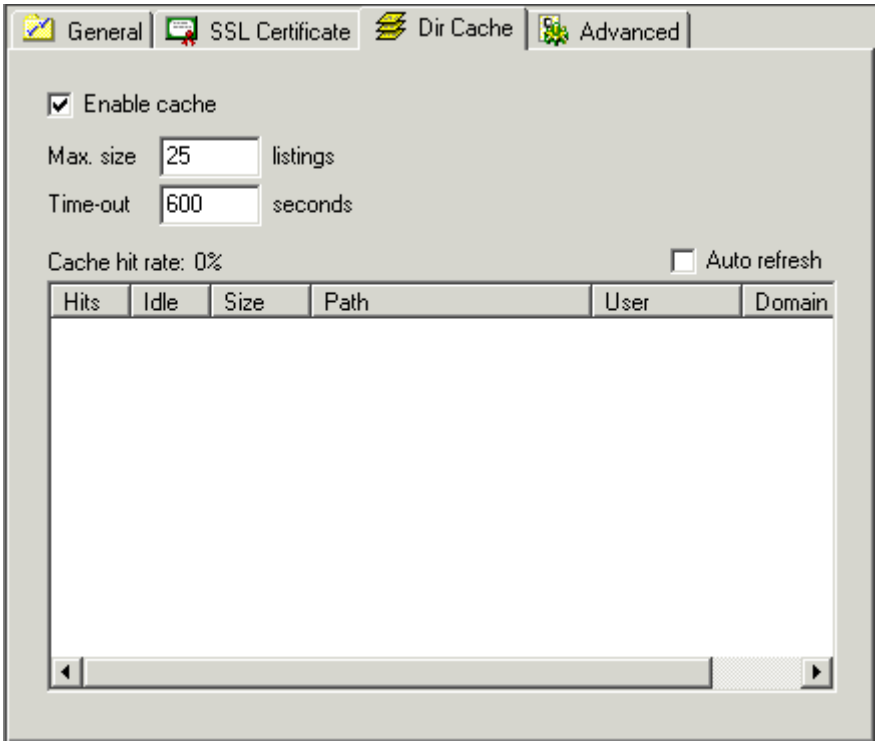
3.4.3. Onglet *Dir Cache*

Ce troisième onglet précise les paramètres de gestion du cache qui garde en mémoire les dernières réponses sollicitées par les requêtes des utilisateurs (commandes du style : *pwd, dir* ou *ls* par exemple).
En effet, créer un listing de répertoire est une opération coûteuse en temps pour le serveur : accéder au disque, vérifier les permissions d'accès de l'utilisateur à chaque élément du listing,

et trier ces éléments. Si un utilisateur demande un listing qui existe déjà dans le cache, le serveur n'aura pas à le recréer, améliorant ainsi grandement les performances du serveur.

Le cache est donc activé par défaut sous Windows 95 et NT avec les paramètres affichés dans la copie d'écran : nombre maximale de 25 listings, délai de conservation en mémoire vive de 600 secondes.

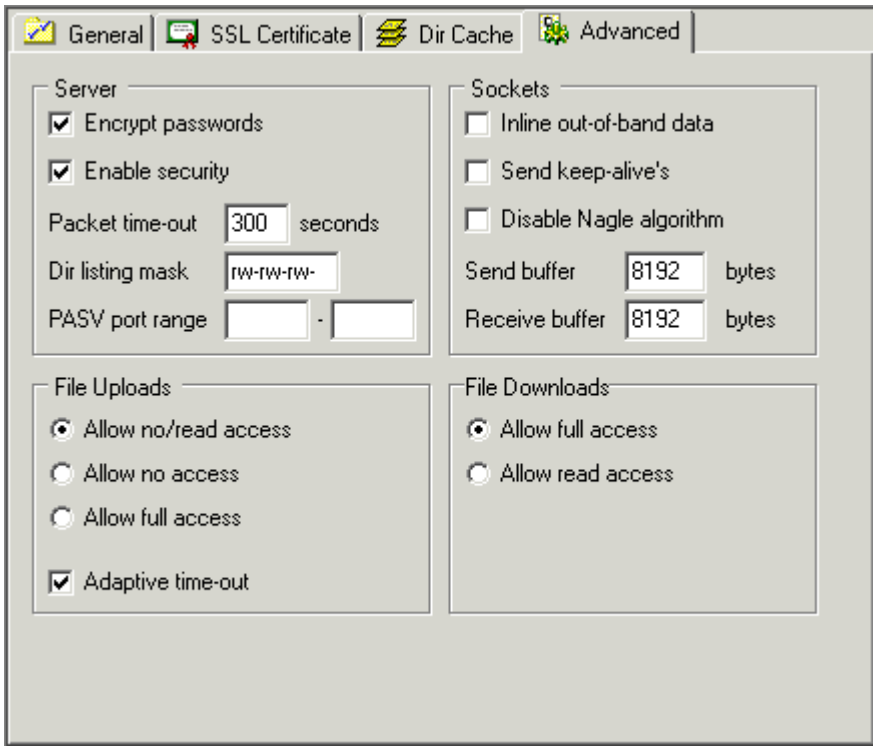
Il offre aussi la possibilité d'afficher la liste des différents listings conservés dans le cache à un moment donné. Il autorise après activation un rafraîchissement en temps réel de ces informations.



3.4.4. Onglet *Advanced*

Ce dernier onglet du noeud *Settings* présente quelques mesures de sécurité disponibles au niveau du serveur, mesures regroupées selon quatre cartouches :

- Le premier cartouche *Server* active par défaut :
  - le cryptage des mots de passe par procédé MD5
    - Les mots de passe sont stockés dans le fichier *ServUDaemon.ini* ou dans le registre.
    - Cette fonction activée assure que personne ne peut lire ou retrouver les mots de passe. Une fois que les mots de passe sont chiffrés, ils ne peuvent plus être déchiffrés. Quand cette case à cocher est désactivée, les mots de passe sont stockés en clair et ils sont visibles dans la configuration du compte d'utilisateur.



- la sécurité avec les paramètres suivants : délai d'acheminement d'un paquet limité à moins de 300 secondes avant de déclarer le transfert avorté, masque proposé pour la visualisation des droits des fichiers.

- Désactiver la sécurité permettrait à n'importe qui du réseau de supprimer/modifier/copier tout fichier du PC ! Il est très fortement conseillé de **ne pas désactiver cette option lors d'une connexion Internet** !

- Il reste à préciser la plage de numéros de port disponibles pour gérer les connexions des clients.

- Par défaut, ce champ reste vierge, ce qui signifie que le serveur utilise un numéro de port aléatoire entre 1024 et 65535 pour chaque transfert en mode passif.

- Le deuxième cartouche *Sockets* règle :

- le cas des données hors ligne (ou hors bande),

- La sélection de cette option permet de repérer les données hors bande parmi les paquets TCP (***paquets urgents***), pour les traiter comme des données normales. Ceci est utile pour compter les **attaques par déni de service** qui envoient de grandes quantités de données hors bande vers des piles de socket qui ne sont plus opérables.

- L'envoi de paquets persistants pour surveiller l'état de la connexion,

- la désactivation de l'**algorithme Nagle**,

- Cela signifie que les paquets sont toujours envoyés dès que possible. Aucun délai supplémentaire n'est introduit, ne serait-ce que l'agrégation des paquets de données sur le réseau (RFC 1122).

- la taille des mémoires tampon en réception et en émission (valeur par défaut sous Windows).

- Le troisième cartouche *File Uploads* fixe :

o les permissions accordées aux autres utilisateurs ou programmes sur les fichiers déposés (lecture seule, pas d'accès ou accès total) pendant leur transfert

o un seuil adaptatif d'avortement du transfert pour permettre à Serv-U de déclarer plus vite un transfert avorté.

- Le quatrième cartouche *File Downloads* montre que les permissions des fichiers téléchargés accordées aux autres utilisateurs ou programmes pendant leur transfert sont limitées à la lecture seule ou à l'accès total (par défaut).

3.4.5. Menus spécifiques

La barre de menus, d'outils et le menu contextuel offrent les options suivantes :

- Pour les onglets *General*, *SSL Certificate* et *Advanced*, deux options qui restent grisées tant que les paramètres ne sont pas modifiés (appliquer les modifications ou restaurer l'ancienne configuration),
- Pour l'onglet *Dir Cache*, les options *Flush* et *Refresh* qui permettent respectivement d'effacer le cache et de rafraîchir l'affichage du cache.

Barre d'outils de tous les onglets sauf l'onglet *Dir Cache*.

Barre d'outils de l'onglet *Dir Cache*.

3.5. Activité du serveur

Comme le montre le volet de gauche, il est possible de surveiller l'activité du serveur aussi bien au niveau du serveur global qu'au niveau d'un site donné.

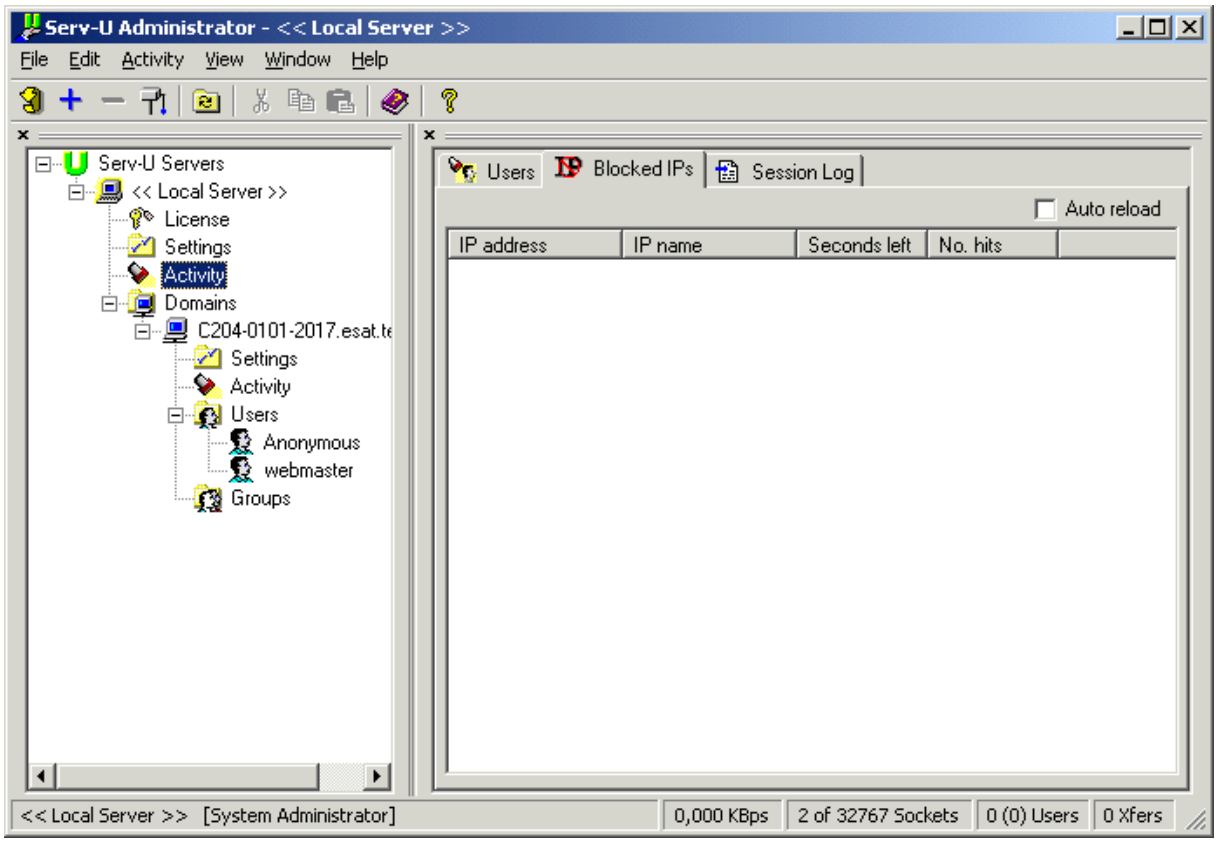
La différence entre les deux niveaux réside dans le nombre d'onglets proposés.

Au niveau du serveur, l'administrateur peut filtrer les clients selon l'adresse IP ou le nom de domaine équivalent selon les procédés habituels aux serveurs Internet reposant sur l'emploi des protocoles TCP/IP.

Les deux autres onglets seront commentés au niveau du paramétrage d'un site.

La barre de menus, d'outils ainsi que le menu contextuel offrent les options spécifiques suivantes :

- Pour l'onglet *Users*, plusieurs options offrent dans l'ordre les possibilités de :
  - rafraîchir l'information (reload),
  - envoyer un message à un utilisateur ou à tous les utilisateurs (broadcast),
  - déconnecter un utilisateur (kill),
  - interrompre un transfert,
  - espionner un utilisateur (spy).



Barre d'outils du nœud *Activity* / onglet *Users*

- Pour l'onglet *Blocked IP's*, la possibilité de recharger la liste des adresses IP des clients bloqués, d'en ajouter d'autres ou d'en supprimer, ou enfin d'effacer la liste,

Barre d'outils du nœud *Activity* / onglet *Blocked IP*

- Pour l'onglet *Session Log*, la possibilité de recharger le contenu du fichier journal, celui de pouvoir filtrer son contenu, ou de copier une partie de son contenu.

Barre d'outils du nœud *Activity* / onglet *Session Log*

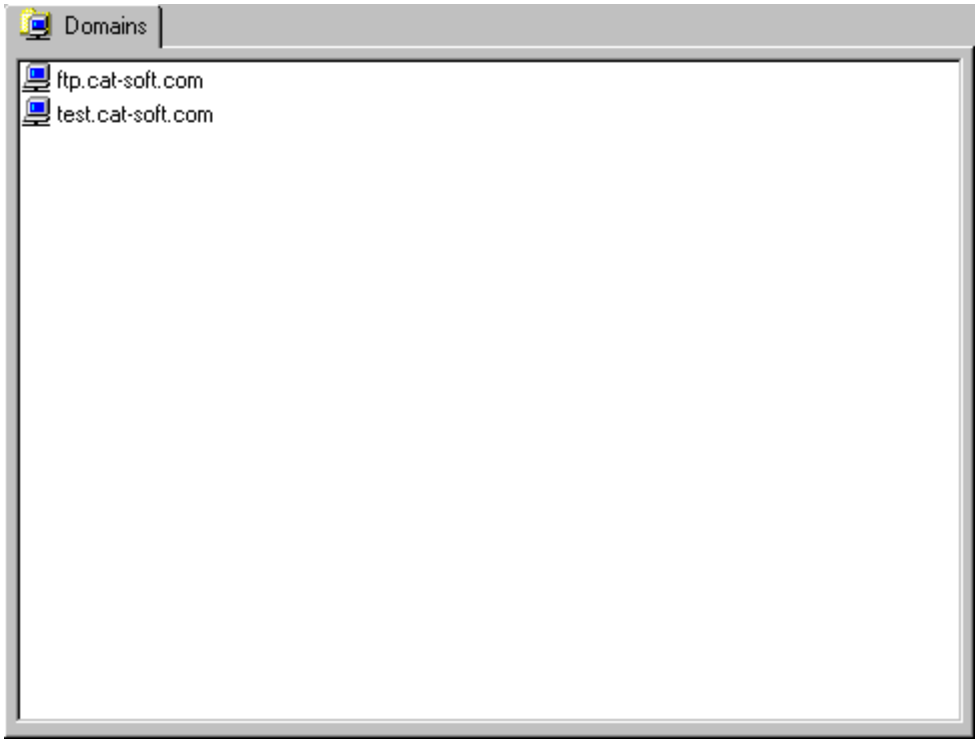
3.6. Domaine d'appartenance du serveur

Il s'agit du quatrième nœud de configuration du serveur qui se décline dans un premier temps selon la station à gérer.

La barre de menus, d'outils ainsi que le menu contextuel offre l'option spécifique suivante de pouvoir créer un nouveau domaine ou de supprimer un domaine existant.

Barre d'outils du nœud *Domaine*.

Un clic droit sur *Domains* dans l'arborescence de gauche offre l'accès au menu *new domain*. La validation de la première boite de dialogue nous permet d'entrer le nom de domaine. Une fois cela fait, une nouvelle arborescence du type de celle mise en place par défaut apparaît.

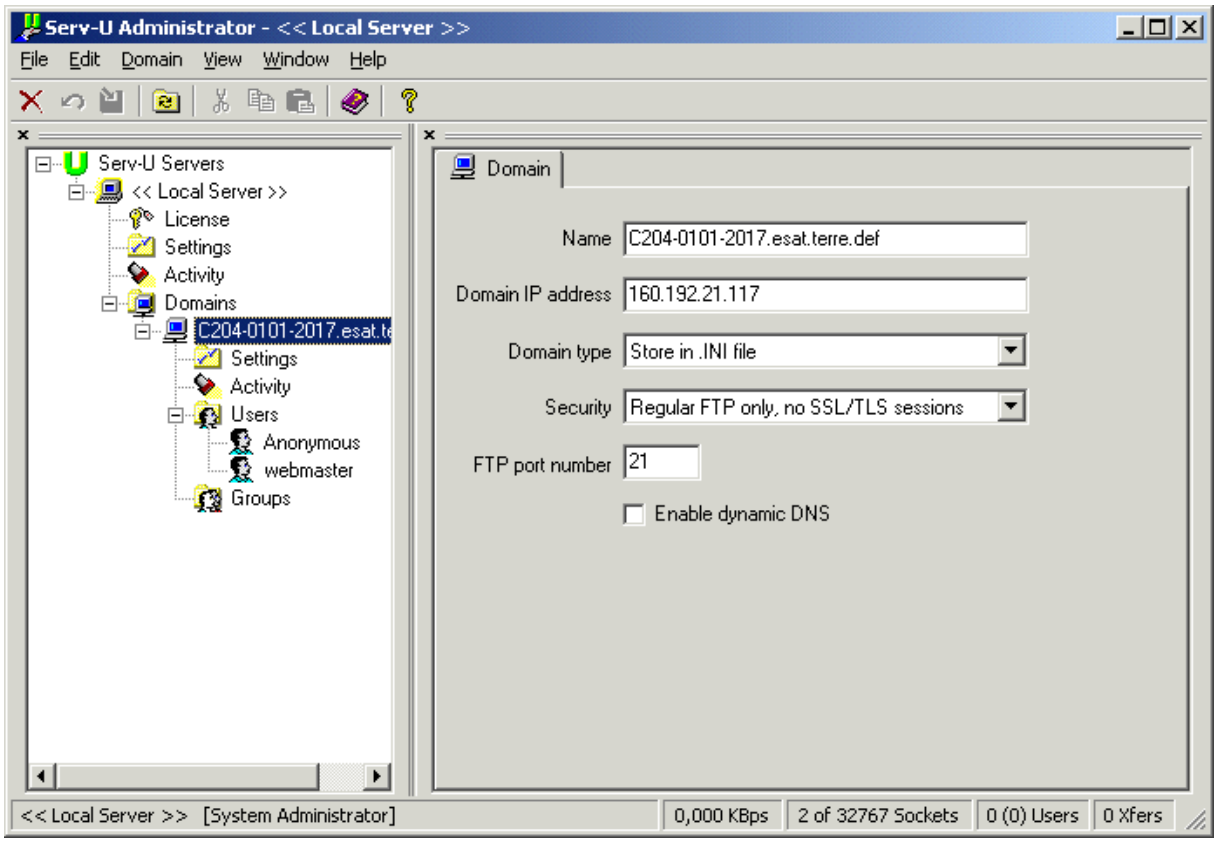


3.6.1. Nœud *Station*

Les identificateurs réseau sont rappelés dans cet onglet :

- Le nom de domaine complet de la station (Name)
- L'adresse IP si elle est assignée au site (ou en blanc pour une adresse IP dynamique),
- Le lieu de stockage du fichier de configuration (Domain type),
  - Le choix d'un fichier *.ini (valeur par défaut) est plus facile d'emploi dans le cas d'un petit fichier de configuration (limite des 64Kb sous Win95/98/Me).
  - Le choix du registre est justifié dans le cas d'un gros fichier de configuration. La clé est : HKEYLOCAL_MACHINE\Software\CatSoft\Serv-U\Domains.
- Le niveau de sécurité employé,
  - La valeur par défaut (Regular …) stipule le non emploi de Secure-FTP (chiffrement de session suivant les techniques SSL/TLS).
  - La 3ème valeur mentionne une utilisation implicite de Secure-FTP.
  - La 2ème valeur autorise les deux modes, moyennant une bascule de l'un à l'autre après demande du client au serveur.
- Le numéro de port de service.

Enfin, la case à cocher *Enable dynamic DNS* nécessite quelques commentaires. En cas de changement d'adresse IP à chaque connexion Internet (type d'abonnement classique ou ADSL), il est possible de consulter un service DNS dynamique (TZO.com) pour obtenir un nom de domaine symbolique comme « votrenom.tzo.com ».



Au niveau de la station, le même menu spécifique *Domain* permet de supprimer un domaine, d'appliquer des modifications ou de restaurer la précédente configuration. Il en est de même de la barre d'outils et du menu contextuel.

Pour chaque station, nous retrouvons les nœuds *Settings* et *Activity* qui héritent naturellement des valeurs des paramètres choisis au niveau du serveur. L'administrateur peut être amené selon les circonstances à modifier les valeurs héritées.

De nouveaux nœuds apparaissent où l'administrateur va distinguer les utilisateurs (Users) selon le type de site mis en œuvre, ainsi que les groupes d'utilisateurs (Groups).

3.6.2. Nœud *Settings*

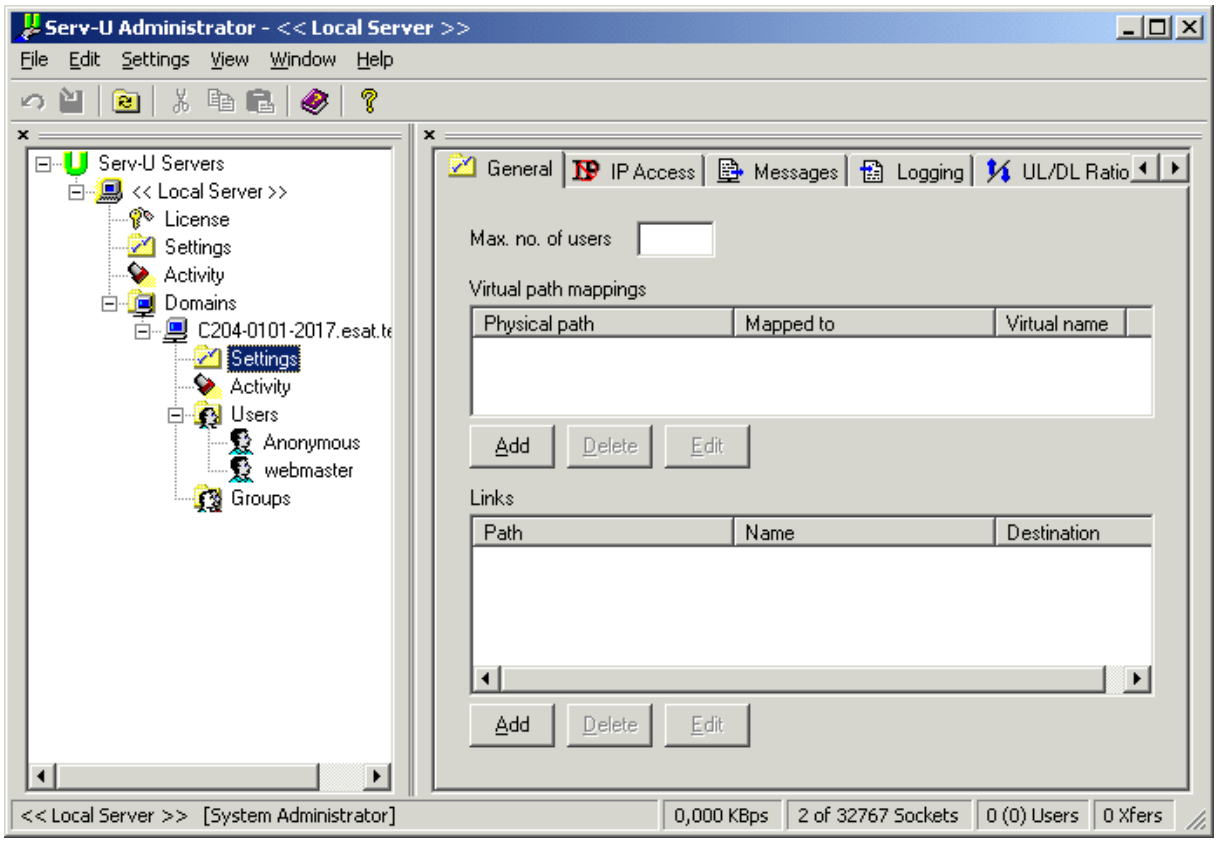
Un clic sur le noeud *Settings* du domaine nous fait découvrir suffisamment d'onglets pour que certains ne soient pas accessibles directement. Il faut utiliser les petites flèches en haut à droite pour se déplacer sur les onglets.

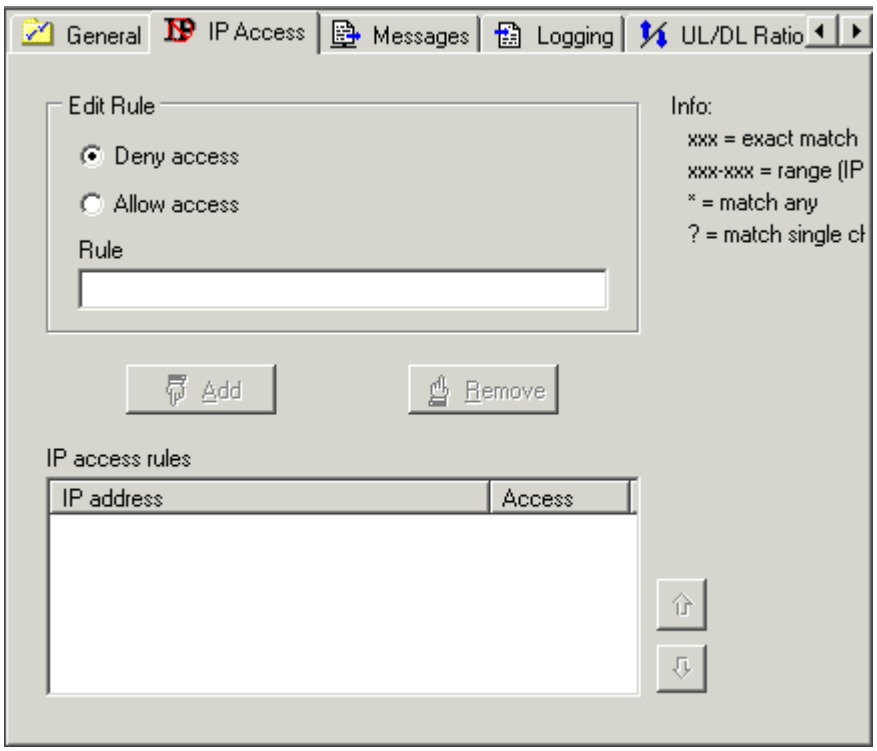
Ce nœud permet de fixer selon les onglets listés dans la colonne de gauche, les paramètres suivants :

General	Fixer le nombre d'utilisateurs de ce domaine qui peuvent être connectés simultanément , et déclarer la liste des répertoires virtuels et des liens.
IP Access	Filtrer à ce niveau les clients et les réseaux indésirables.
Messages	Configurer les messages liés à la connexion au serveur.
Logging	Décider du contenu des fichiers journaux
UL/DL Ratios	Fixer les contraintes pour la dépose de fichiers ainsi que les quotas de disque, ceci par utilisateur.

Advanced	Décider l'emploi de paramètres supplémentaires.

a. Onglet *General*





b. Onglet IP Access

Cet onglet (copie d'écran précédente) comprend deux cartouches :

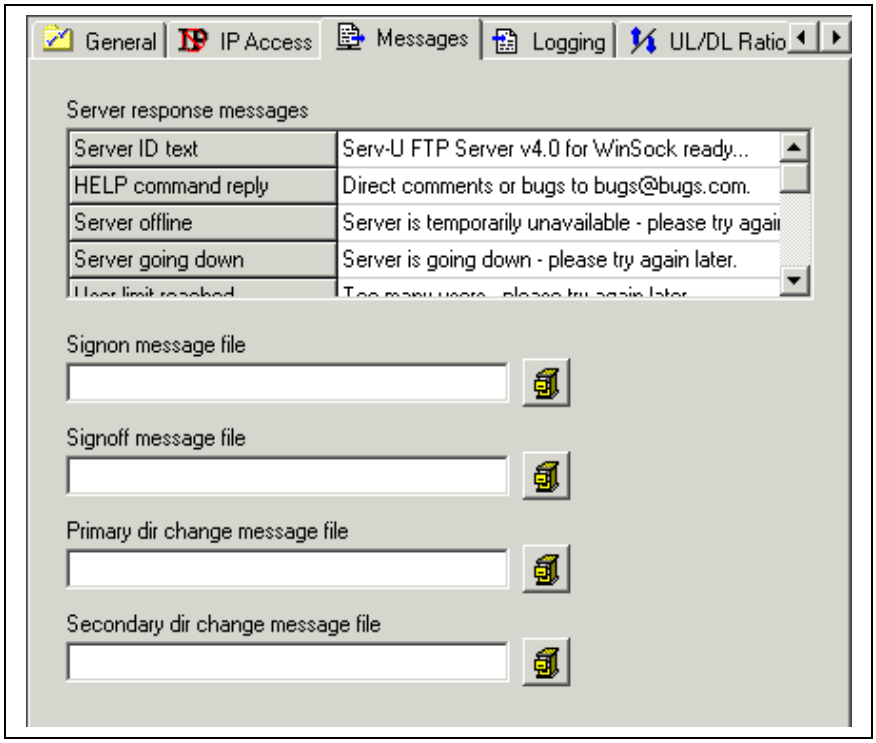
- Un pour définir la règle prioritaire (deny par défaut), celle d'interdiction (deny) ou d'autorisation (allow) par choix d'un bouton radio, puis un champ pour définir une par une quelles sont les adresses concernées par cette règle ;

- Un cartouche pour afficher la liste des exceptions à cette règle.

Les différentes syntaxes possibles sont rappelées dans la marge de droite.

c. Onglet *Messages*

Cet onglet présente d'abord un cartouche où il est possible de personnaliser les réponses du serveur au client quand ce dernier se connecte au serveur en mode commande.



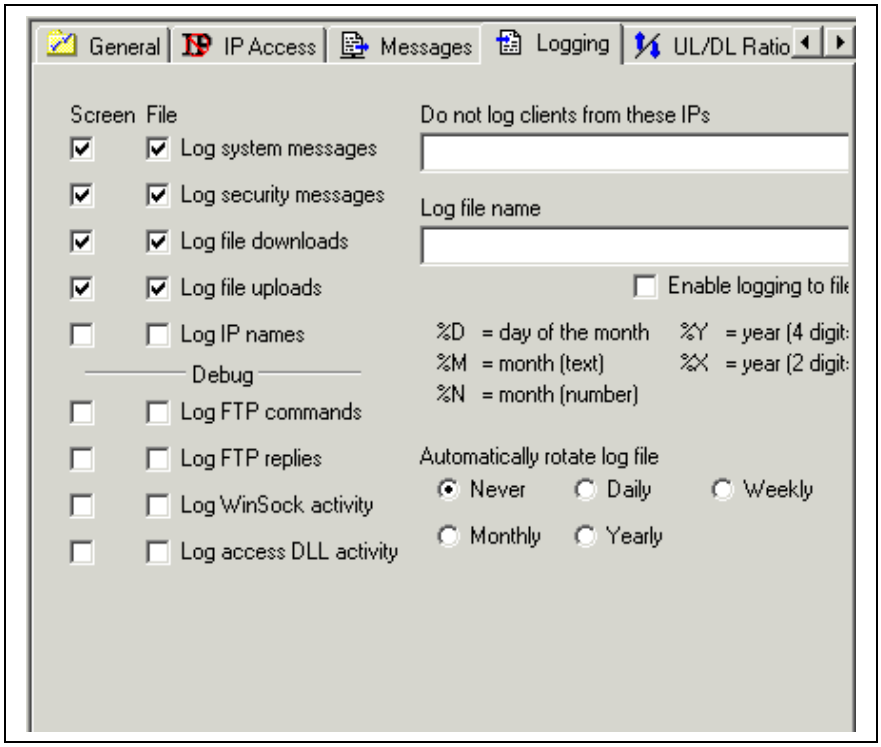
Server response msg	Fixe les différents messages renvoyées aux clients selon les actions effectuées par ce dernier, ou événements affectant le comportement du serveur : • nom du serveur, • commande d'accès à l'aide en ligne, • message dans le cas d'indisponibilité du serveur, • message dans le cas de l'arrêt du serveur, • message dans le cas du nombre d'utilisateurs maximum atteint, • message dans le cas d'un accès non anonyme.
Signon message ...	Sélectionne un fichier .txt qui contiendra le message d'accueil du serveur ftp.

Signoff message …	Sélectionne un fichier .txt qui contiendra le message de sortie du serveur ftp.
Server dir change …	Renvoie au client les informations liées aux actions de changement de répertoires.

d. Onglet *Logging*

Cet onglet commence par présenter à gauche deux colonnes de cases à cocher. L'administrateur peut ainsi distinguer les types d'informations à afficher à l'écran et ceux à sauvegarder dans les fichiers journaux. Il peut cocher les deux cases pour chaque type d'informations, à l'image de la copie d'écran ci-dessous (options par défaut).

Les deux premières lignes concernent les fichiers journaux du système d'exploitation Windows (système et sécurité).
Les deux lignes suivantes concernent la journalisation des transferts.
Les autres lignes permettent de rajouter des informations dans ces fichiers journaux.



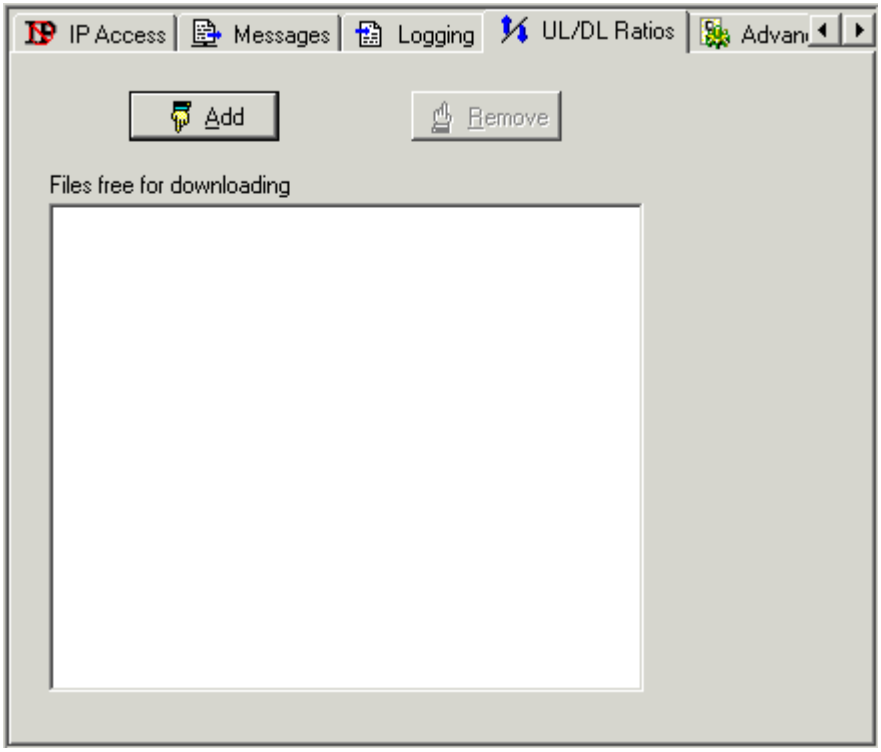
Le premier champ texte de droite offre la possibilité de ne pas enregistrer l'activité de certains clients (administrateurs par exemple pour ne pas fausser les statistiques d'activité).

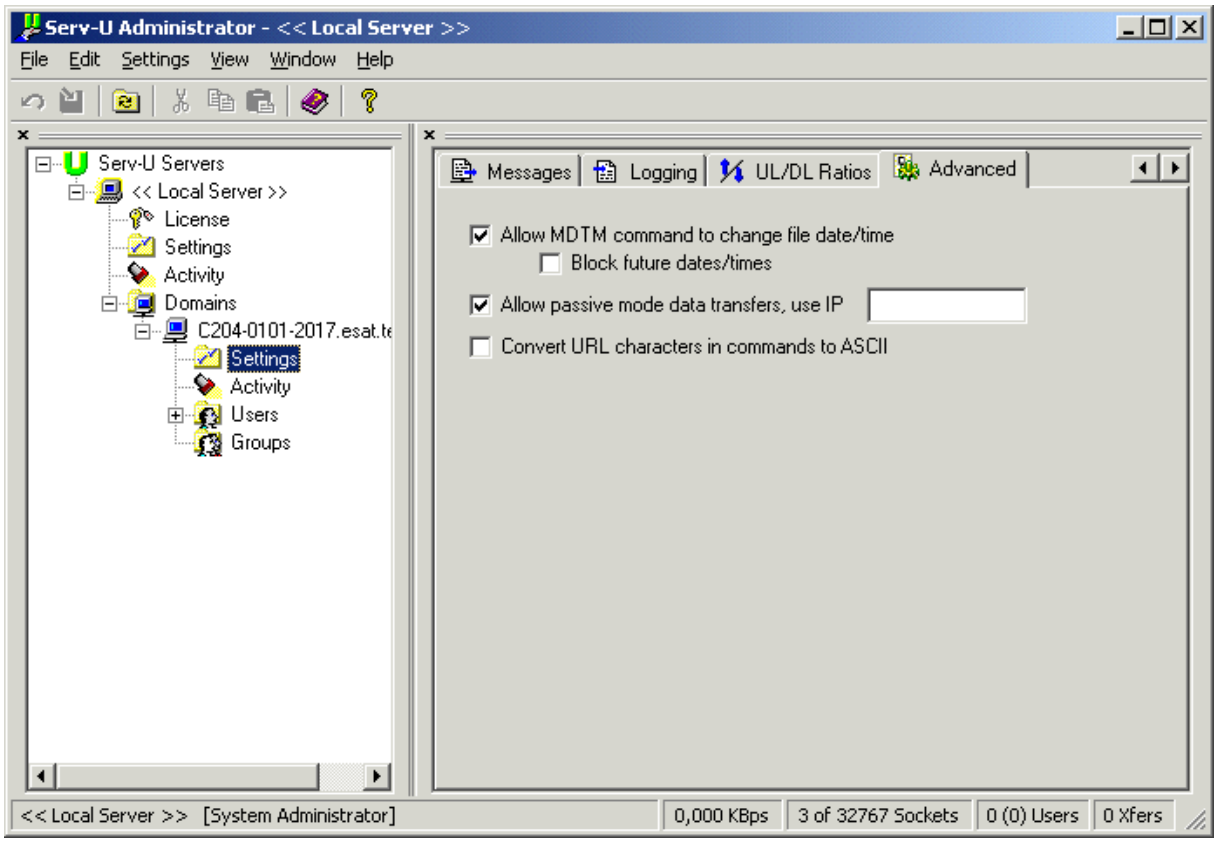
L'administrateur doit ensuite décider de la périodicité de changement de fichiers journaux selon la fréquentation estimée du serveur. Le choix par défaut (Never) n'est pas conseillé.

Un serveur d'activité modeste peut choisir une périodicité mensuelle. Dans ce cas, pour s'y retrouver plus aisément, il peut donner un nom précis à chaque instance de fichier journal.
Il doit d'abord activer cette fonction à l'aide de la case à cocher située juste en dessous avant de préciser le nom des fichiers journaux grâce à la syntaxe rappelée en dessous (*%M%X.log* par exemple).

e. Onglet *UL/DL Ratios*

Cet avant-dernier onglet permet de définir les fichiers libres en nombre de transferts aussi bien pour les téléchargements (DL) que pour les déposes (UL).





f. Onglet *Advanced*

Cet onglet (copie d'écran précédente) permet de choisir trois options :

- autoriser les ***commandes MTDM*** à modifier les dates et heures des fichiers du site, puis bloquer tout nouveau changement ;
  - Ceci concerne les utilisateurs autorisés en écriture sur le site.
- autoriser des transferts de fichiers en mode passif sut telle adresse IP ;
  - Ce champ est à renseigner dans le cas où un serveur proxy ou parefeu se situe entre le réseau Internet et le serveur FTP.
- convertir les caractères des adresses URL en commandes.
  - Les caractères spéciaux présents dans les adresses URL, tels que « %20 » pour représenter le caractère espace, sont effectivement remplacés par le code correspondant.

3.6.3. Noeud *Activity*

Au niveau du nœud *Activity* de la station, les onglets et paramètres offerts à ce niveau sont les même que ceux décrits au niveau du serveur local.
Seul l'onglet *Session Log* offre au niveau de la barre de menus, la barre d'outils et du menu contextuel, la possibilité supplémentaire d'effacer le contenu.

3ème icône de la barre du noeud *Activity* / onglet *Session Log*.

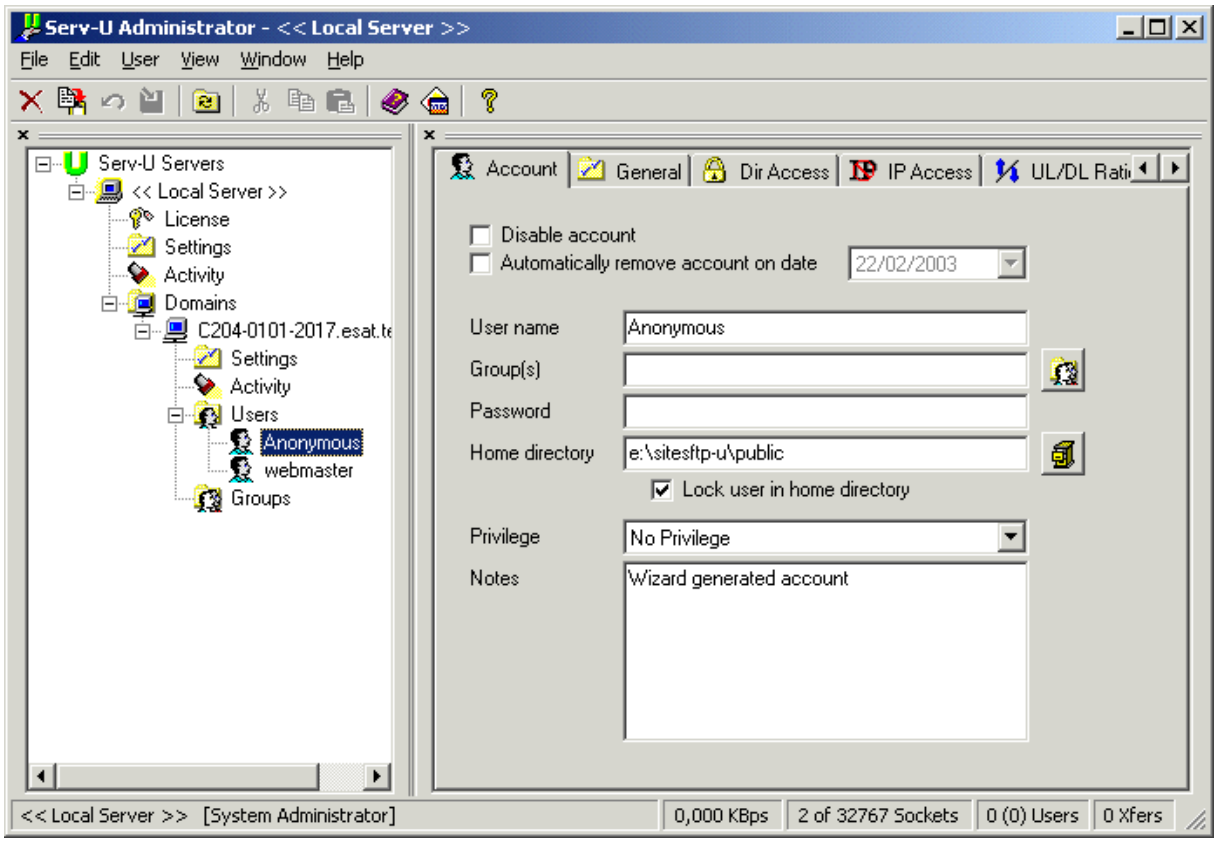
3.6.4. Nœud *Users*

Le domaine est maintenant créé et paramétré. Il reste à configurer l'accès des utilisateurs.

La sélection du nœud *Users* n'apporte aucun paramétrage supplémentaire.

Le choix du nœud ***Anonymous*** fait apparaître six onglets dans le volet de droite :

- un onglet *Account* pour gérer le compte en question,
- un onglet *General* pour définir quelques caractéristiques supplémentaires sur les accès, la durée de la connexion, les vitesses de transfert, etc…
- un onglet *Dir Access* pour poser les permissions sur le site en question au niveau des fichiers et répertoires,
- un onglet *IP Access* pour filtrer les accès des clients selon leurs identifiants réseau (onglet déjà commenté)
- un onglet *DL/UL Ratios* pour limiter le nombre de transferts des clients,
- un onglet *Quotas* pour limiter les sites en quotas de disque.



a. Onglet Account

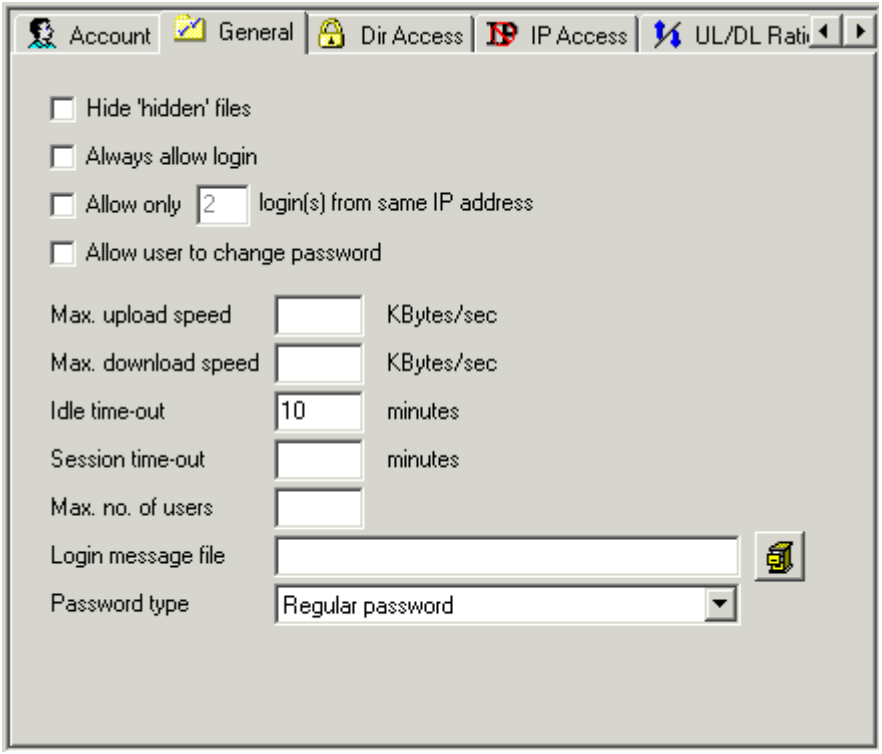
Cet onglet offre la possibilité de fixer la durée de vie du compte : le désactiver ou assortir la vie du compte d'une durée de validité.
Les champs suivants rappellent :

- les caractéristiques d'accès au compte (nom d'utilisateur, groupe d'appartenance, mot de passe),
  - Pour un site privé où le changement de mot de passe périodique est nécessaire, cette opération s'effectue directement dans le champ texte prévu à cet effet.
  - Le mot de passe est indiqué en clair, puis le bouton *Apply* de la barre d'outils est activé pour chiffrer ce dernier et rendre le champ opaque.
- la localisation de la racine du site (home directory),
- les permissions posés sur ce compte (verrouillage du compte au niveau de la racine du site, privilège d'administrateur ou non).

b. Onglet *General*

Hide hidden files	Cache effectivement les fichiers cachés.
Always allow login	Autorise cet utilisateur à pouvoir se connecter même dans le cas où le seuil du nombre de connectés est atteint.
Allow only … login(s)	Autorise X connexions par adresse IP (procédé de *multithreading*) (1).
Allow user to ...	Autorise l'utilisateur à changer de mot de passe (***pas conseillé*** !).

Max upload speed	Vitesse maximum de dépose.
Max download speed	Vitesse maximum de téléchargement.
Idle time out	Temps d'inactivité autorisé avant de déconnecter telle adresse IP.
Session time out	Temps que l'utilisateur peut rester connecté.
Max no. of users	Fixe le nombre d'instances d'un compte d'utilisateur donné à un moment « t ».
Login message file	Personnalise le message d'accueil pour ce compte d'utilisateur.
Password type	Fixe le type de mot de passe de l'utilisateur, et donc le type d'authentification.



(1) ***multithreading***
Ce procédé consiste à se connecter plusieurs fois au même endroit pour optimiser sa connexion.

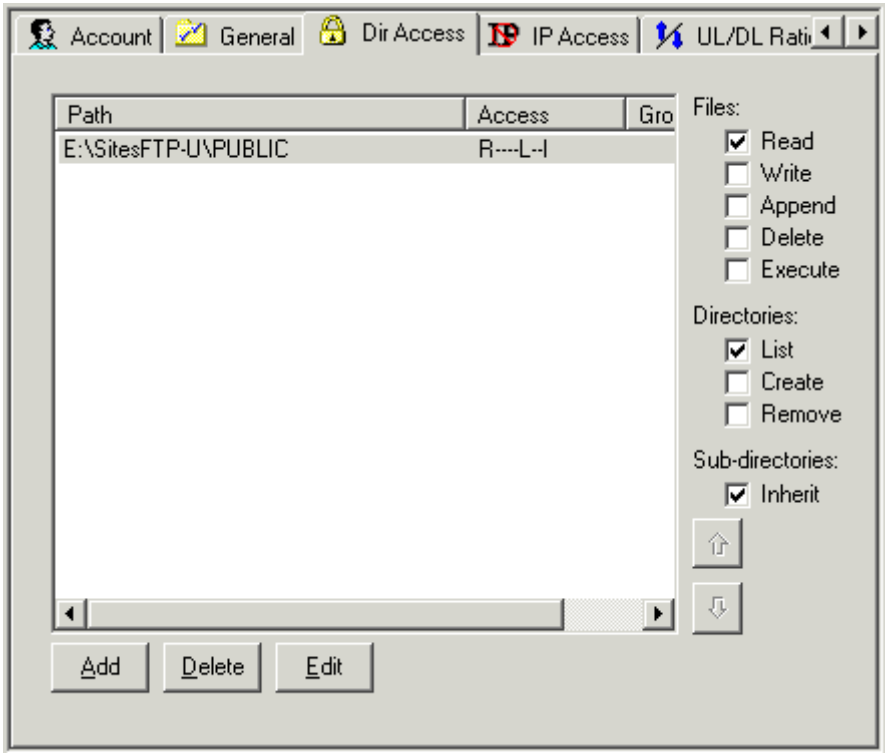
c. Onglet *Dir Access*

Nous finissons par définir à l'utilisateur un dossier d'accès.

Dans l'onglet *Dir Access*, nous effectuons un clic droit dans la zone d'action puis nous choisissons *Add Rules* ou cliquons sur le bouton *Add* situé en dessous de cette zone.
Nous définissons alors le chemin du dossier de cet utilisateur, puis s'il a le droit de voir ou non le chemin d'accès physique (ex: E:\SitesFTP-U\PUBLIC\).
Une fois toutes les actions effectuées, le dossier apparaît dans la zone d'action. Nous pouvons désormais paramétrer les permissions que cet utilisateur a sur ce dossier.

Read	Autorise le téléchargement ou *download* (par défaut).
Write	Autorise la dépose (upload).
Append	Autorise le compte-rendu de dépose.
Delete	Autorise la suppression de fichiers.

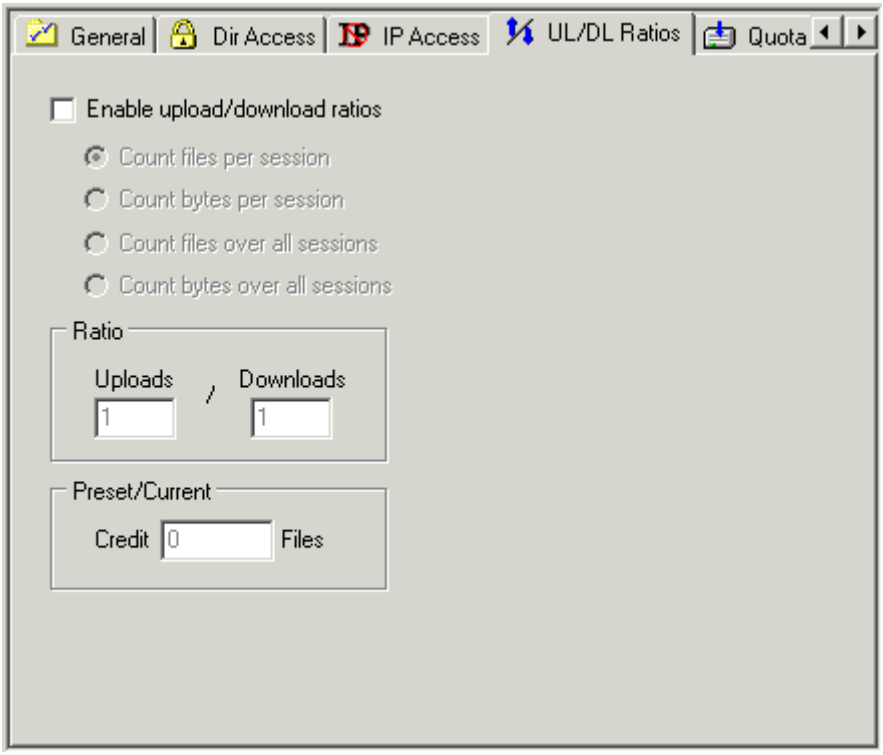
Execute	Autorise l'exécution d'un fichier **(en général INTERDIT !)**
List	Autorise le listage du contenu du dossier.
Create	Autorise la création de dossiers.
Remove	Autorise la suppression de dossiers.
Inherit	Les droits de ce dossier sont appliqués sur l'ensemble des fichiers/dossiers enfants de niveaux inférieurs (sous-dossiers).



Un changement de droit peut se faire en dynamique sans arrêter le serveur. Il suffit de cocher les nouveaux droits, puis d'activer le bouton *Apply* de la barre d'outils.

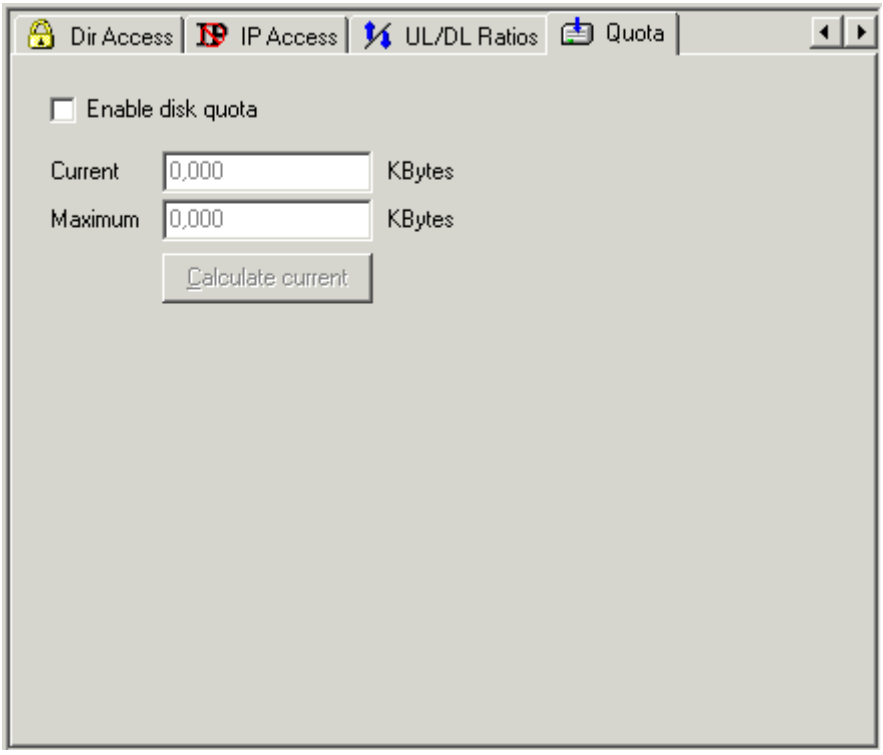
d. Onglet *DL/UL Ratios*

Cet onglet a pour but de donner, à tout le site, des règles générales pour limiter le nombre de transferts des clients selon plusieurs critères possibles : fichiers ou octets, par session ou pour toutes les sessions, le nombre en question, etc...



e. Onglet *Quotas*

Ce dernier onglet permet de donner des limites sur les sites en quotas de disque, notamment en matière de dépose.



La barre de menus, la barre d'outils et le menu contextuel offrent au niveau des nœuds *Users* ou *Groups*, la possibilité de créer un nouvel utilisateur, de le supprimer ou de copier son contexte.

Les trois premières icônes de la barre d'outils *Users* et *Groups* répondent à ce besoin.

3.6.5. Nouveau compte

Pour créer un nouveau compte, il faut :

- choisir dans le menu contextuel du composant *Users* l'option *Add New User*
- entrer le nom de l'utilisateur (login) puis son mot de passe (password) séparé par le caractère « : », par exemple : nadia:saketeam.

Annexe 1 : Administration locale KDE de Wu-FTP

A priori, la société qui fournit l'environnement graphique KDE au profit de la distribution LINUX Mandrake a intégré dans la liste des logiciels fournis celui du serveur FTP *wu-ftpd*, qu'il a renommé pour l'occasion **kwuftpd**.

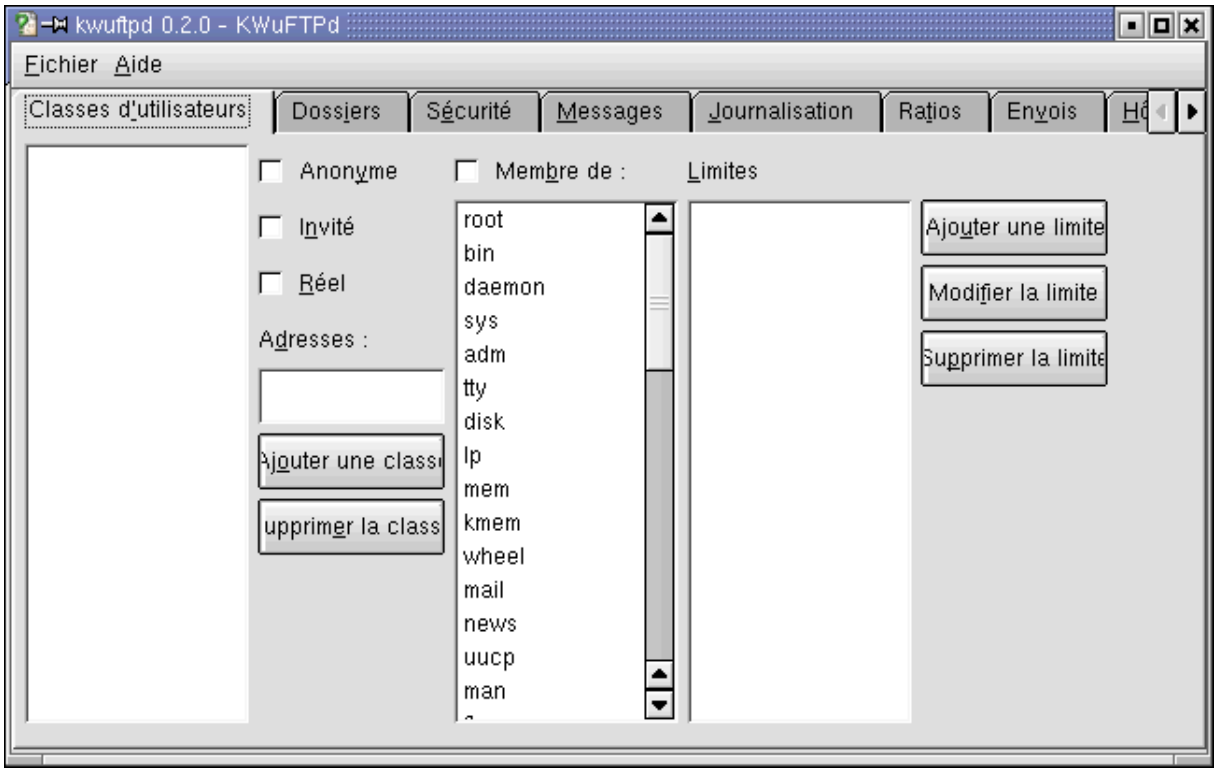
Il est donc présent dans le menu *Démarrer* de cet environnement. Pour le démarrer en mode console, il suffit de lancer la commande *kwuftpd*.

L'administration graphique de ce serveur nous est proposé à travers le parcours de huit onglets de configuration.

Convention d'écriture : dans toutes les annexes, les commentaires rappellent entre parenthèses et en gras le nom de la directive correspondante à mettre en œuvre dans le fichier de configuration du serveur étudié.

1. Classe d'utilisateurs

Le premier onglet permet de définir des classes d'utilisateurs (**class**) en fonction des trois types inventoriés (anonyme, réel ou invité). Pour chaque classe, nous précisons l'adresse du réseau d'appartenance.

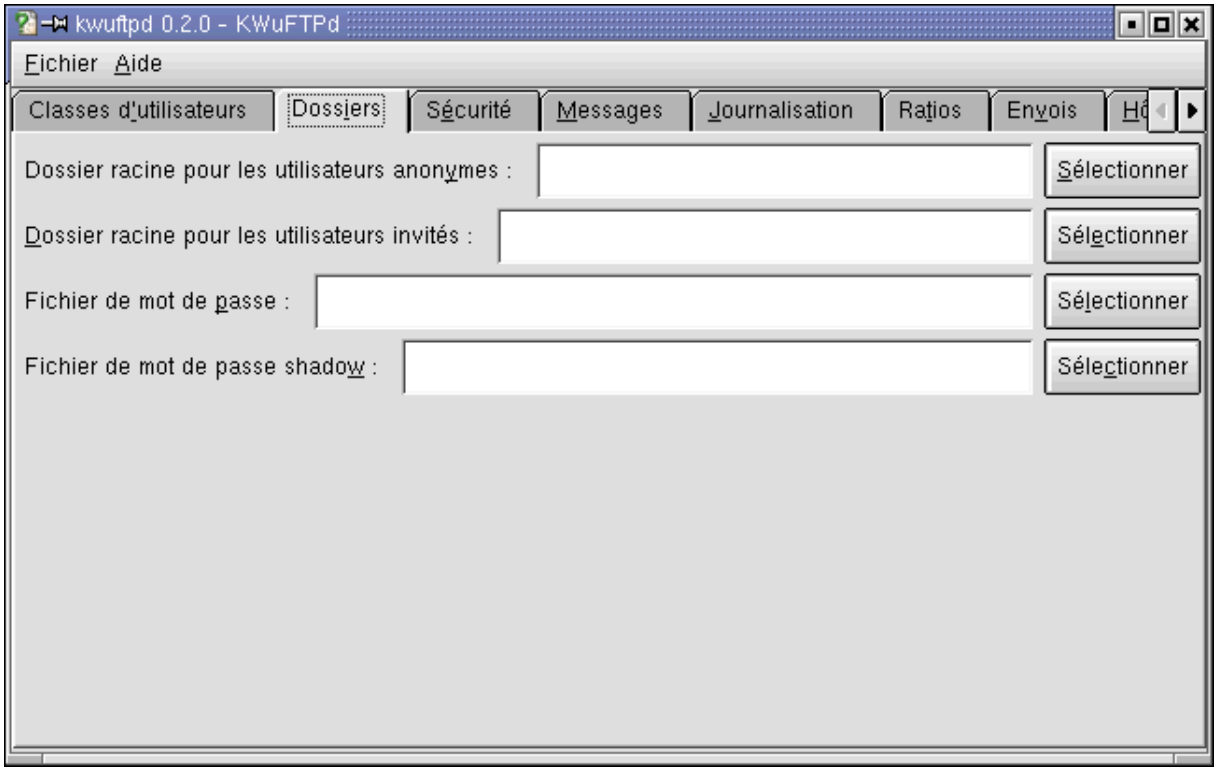


Les utilisateurs invités sont déclarés au niveau du système (**guestuser**, **guestgroup**) et donc présents dans la liste de droite.

Pour finir, nous pouvons poser des limites en nombre d'utilisateurs connectés (**limit**).

2. Dossiers

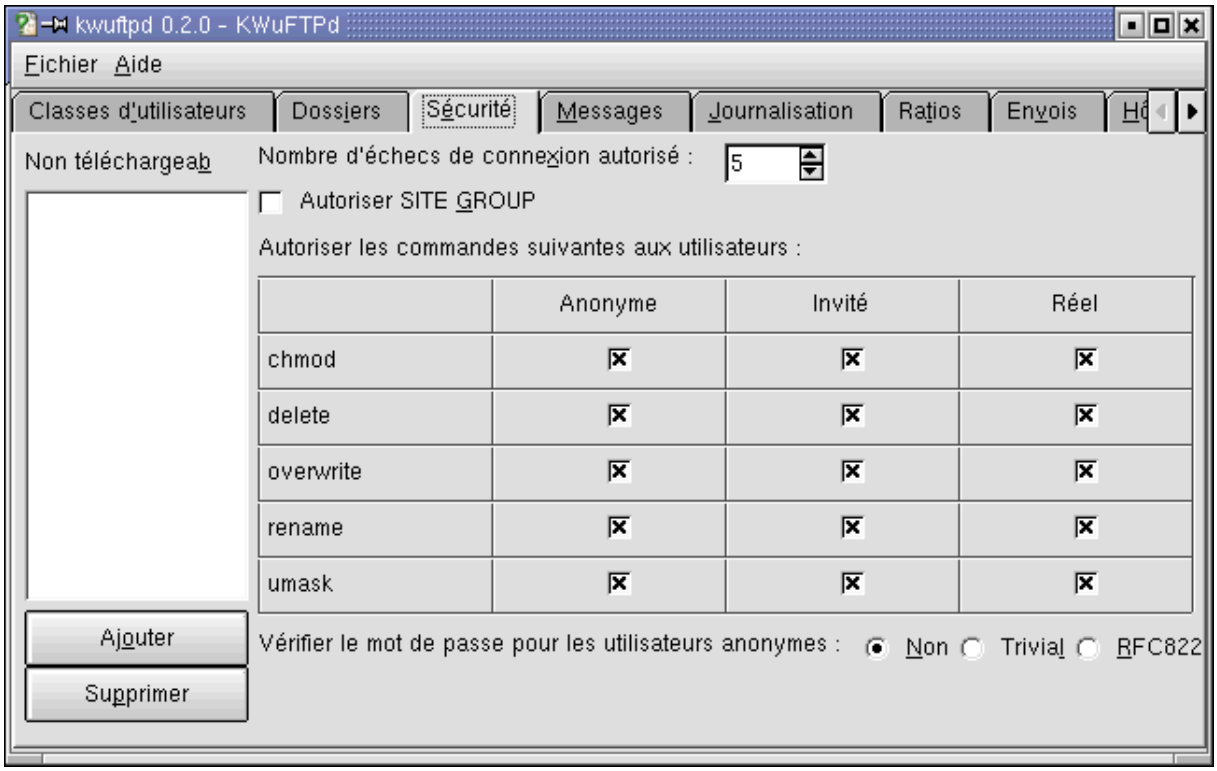
Le deuxième onglet permet de choisir le dossier racine des sites destinés aux utilisateurs anonymes (site public) ou aux utilisateurs invités (site privés).
Des fichiers de mots de passe spécifiques au service FTP peuvent être définis ici.



3. Sécurité

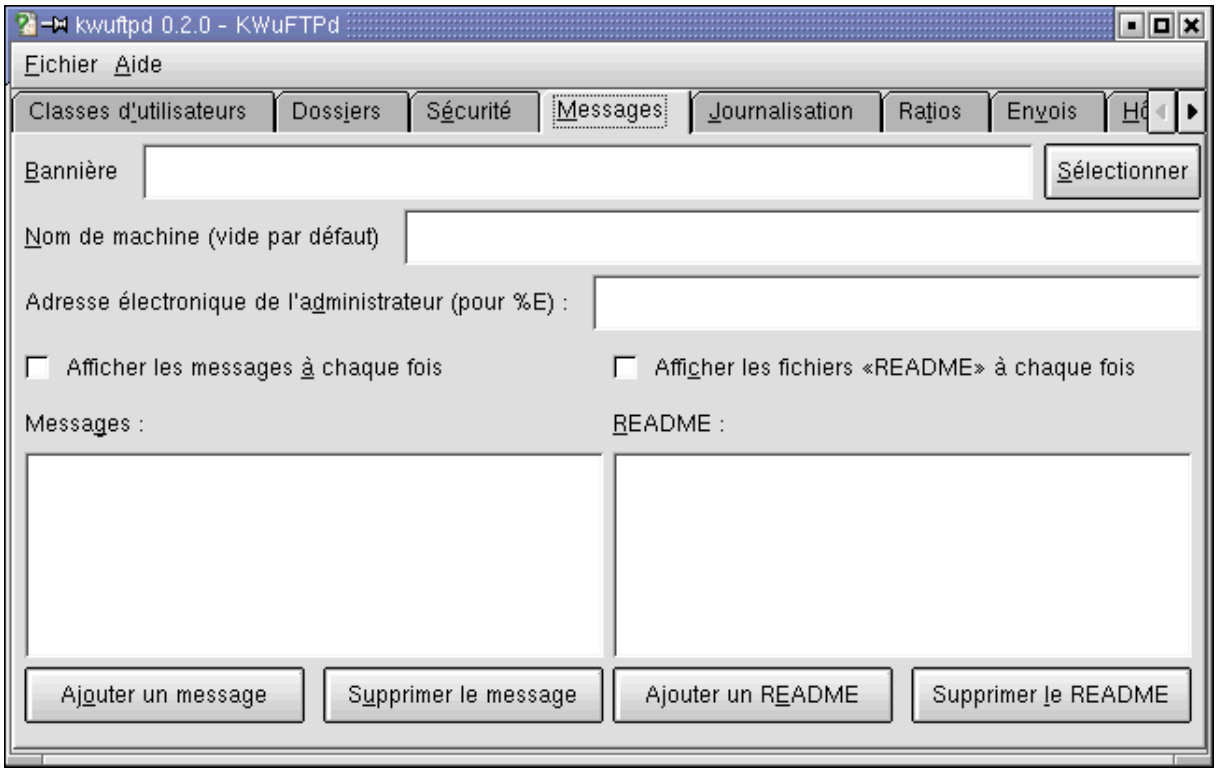
Le troisième onglet permet de définir :

- les opérations permises sur les sites,
- le nombre de tentatives de connexion (**loginfails**) ainsi que le mode de vérification de mots de passe (**passwd**) des connexions anonymes (trois choix possibles),
- la liste des dossiers ou fichiers non téléchargeables.



4. Messages

Le quatrième onglet précise les informations renvoyées au client : fichier de bannière d'accueil (**banner**), nom du site ou serveur (**hostname**). Il précise aussi les fichiers dont ils sont extraits.



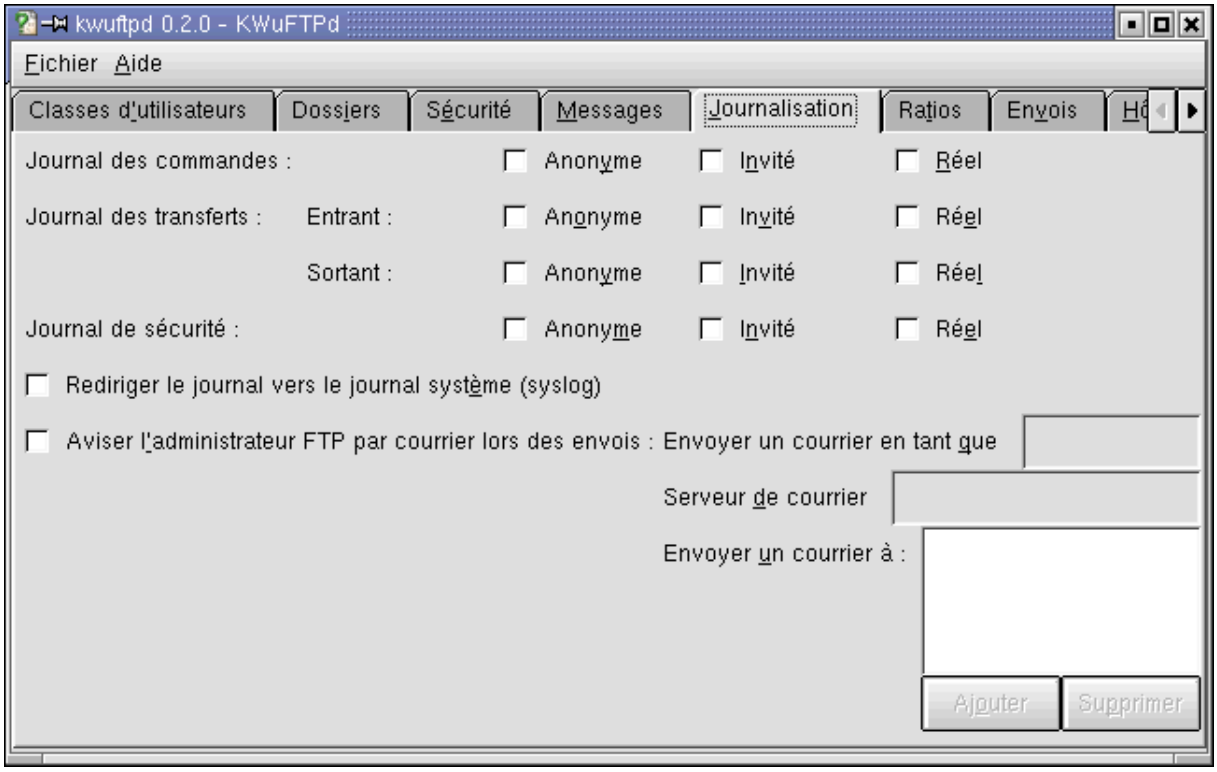
Nous pouvons par exemple renseigner l'adresse de courriel de l'administrateur (**email**) si celle-ci est mentionnée dans les messages renvoyées.

L'activation des cases à cocher « Afficher … à chaque fois » fait que le serveur continue d'afficher les messages à chaque passage (**message, readme**).

5. Journalisation

Le cinquième onglet configure le(s) fichier(s) journal(ux) utilisé(s) (**log**) : leur nombre, les gestionnaires de ces fichiers (syslog ou non), les types d'utilisateurs surveillés, le type de transfert.

L'administrateur peut en renseignant les trois champs exposés (**mailfrom, mailserver, email**) être avisé par message de tout événement.

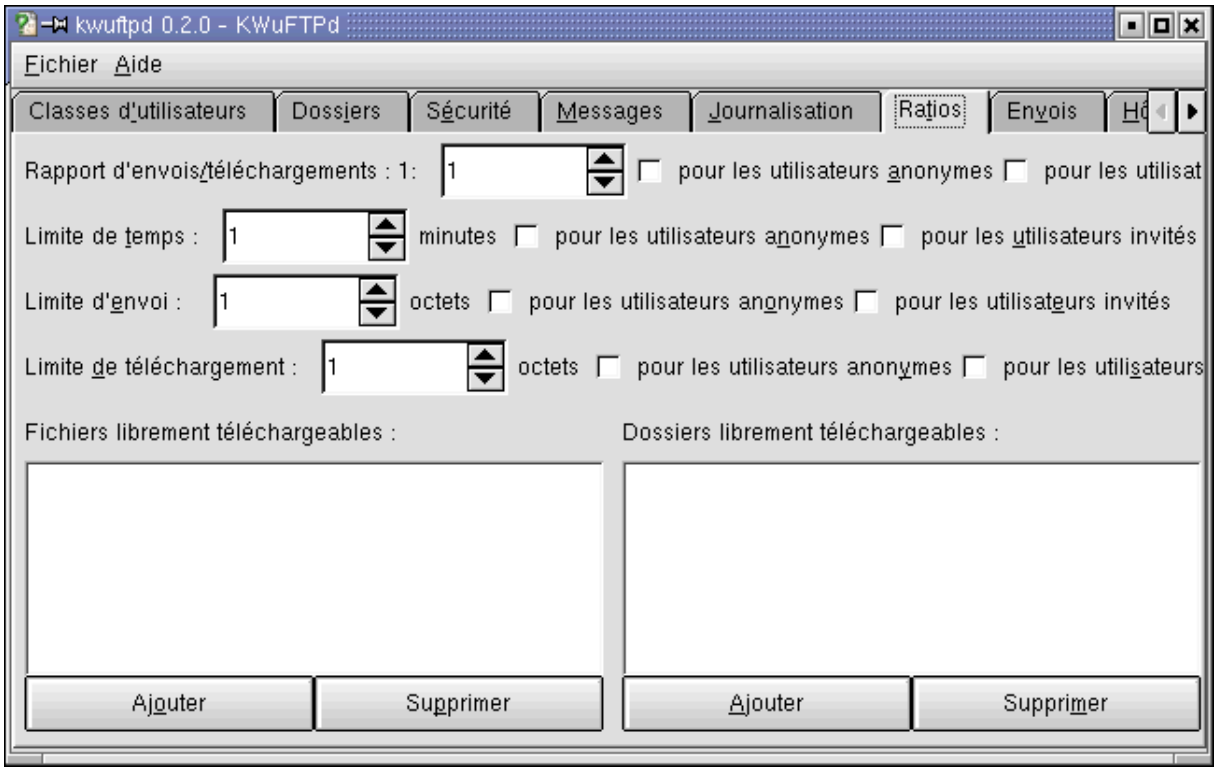


6. Ratios

Le sixième onglet entretient la liste des fichiers et dossiers libres de transferts (**dl-free, dl-free-dir**).

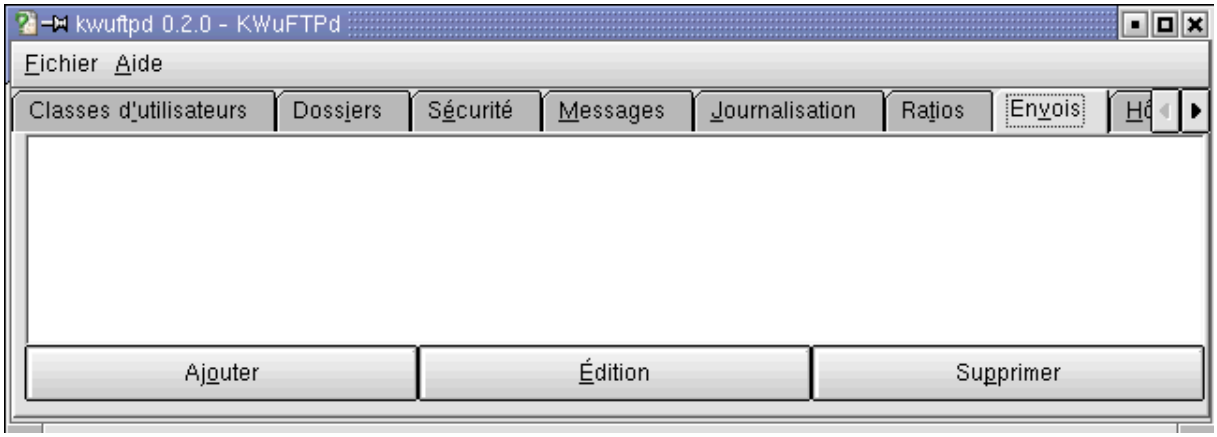
Il restreint aussi les droits des utilisateurs dans :

- le nombre de transferts (**ul-dl-rate**),
- les délais de session et de transferts (**timeout…**),
- les quantités d'octets envoyés ou téléchargés (**data-limit, file-limit**).



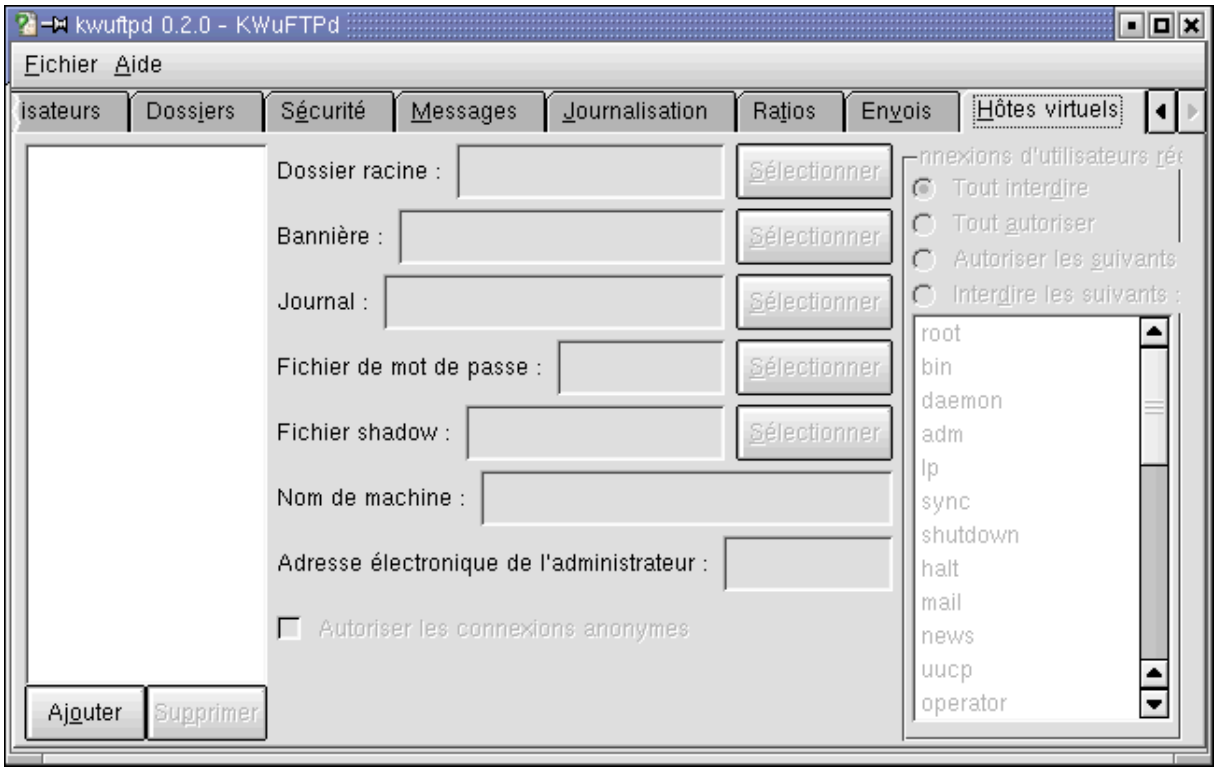
7. Envois

Le septième onglet surveille les transferts.



8. Hôtes virtuels

Le huitième onglet configure des hôtes virtuels (**virtual**) : sites autres que le site par défaut.



Nous pouvons indiquer le nom du site dans le champ « Nom de machine ».
Cet onglet nous apprend que nous pouvons personnaliser la bannière d'accueil, la gestion des fichiers journaux, les fichiers des mots de passe, et l'adresse de courriel de l'administrateur.

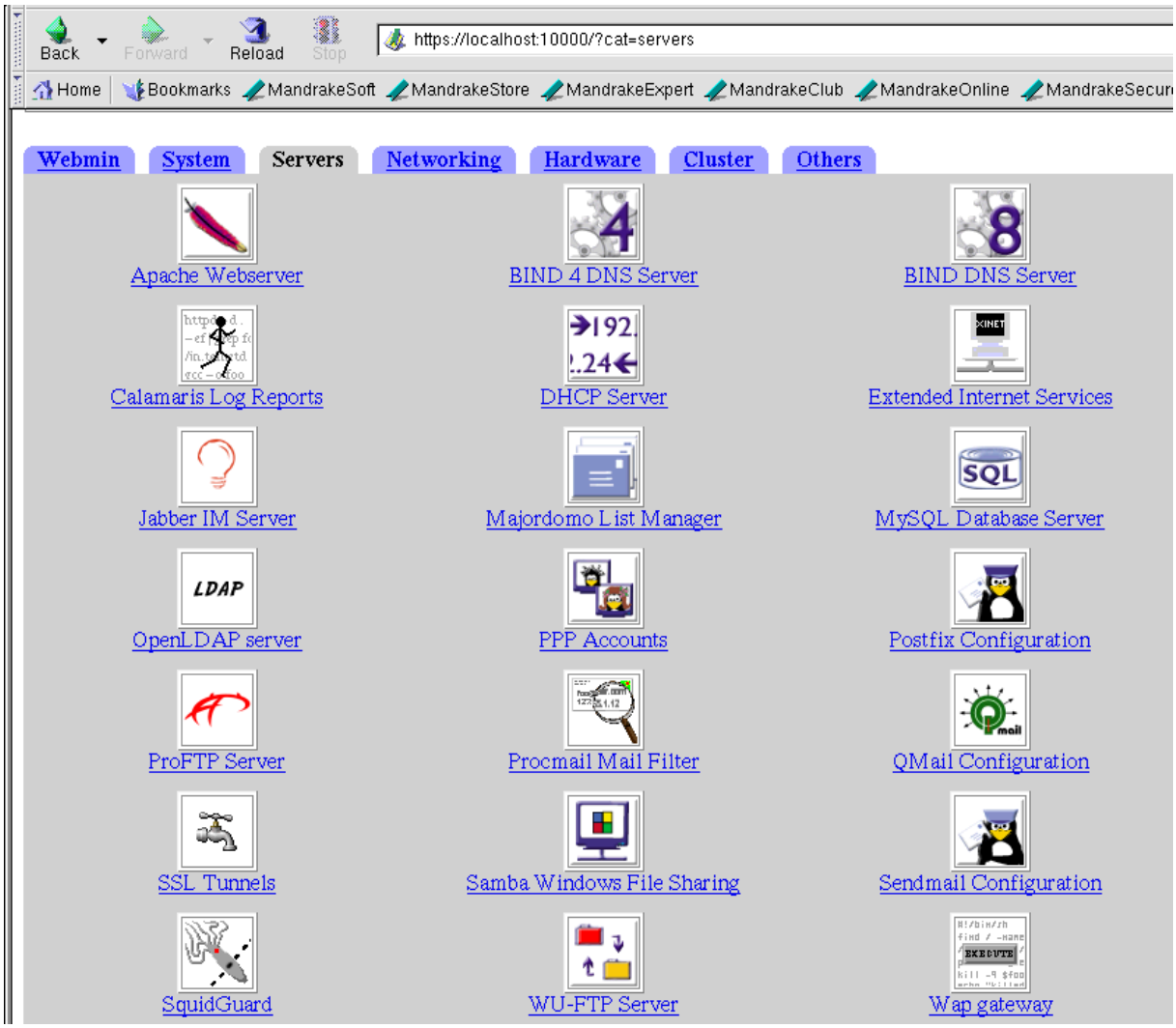
Annexe 2 : Administration distante Webmin pour Wu-FTP

Le document présent suppose que le programme Webmin, issu du monde des logiciels libres, ait été installé sur la station hébergeant le serveur FTP, ainsi que le serveur Wu-ftp lui-même.

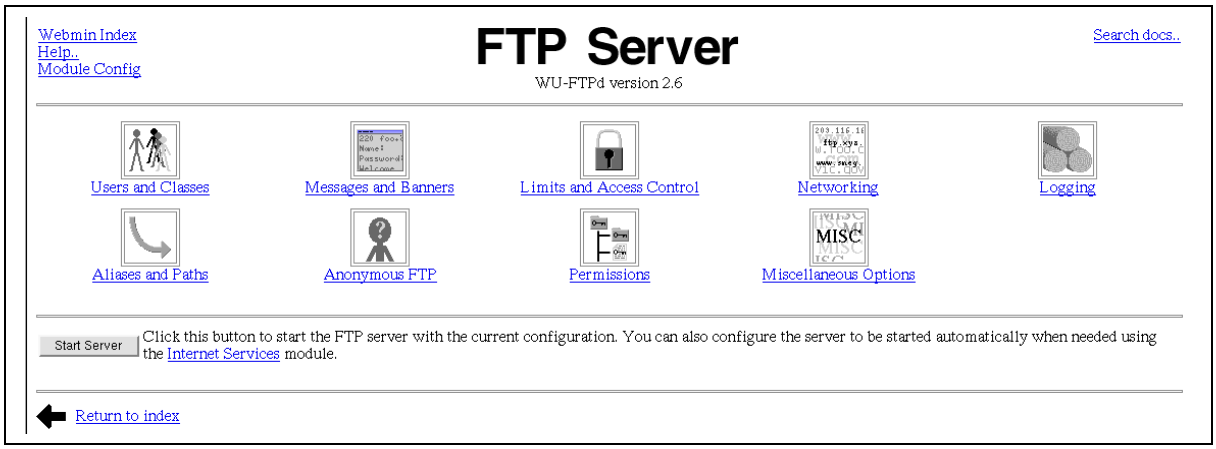
1. Menu général

Il est rappelé que ce logiciel permet l'administration à distance aussi bien du système que celui des services Internet, le tout via un interface Web.
Vu la gravité des sujets traités, son accès se fait via une connexion sécurisée (https) sur un numéro de port spécifique (10.000).

Une fois l'onglet *Serveurs* sélectionné, nous trouvons en bas de la liste le serveur *Wu-ftp* (voir écran suivant), menu-icône que nous activons.

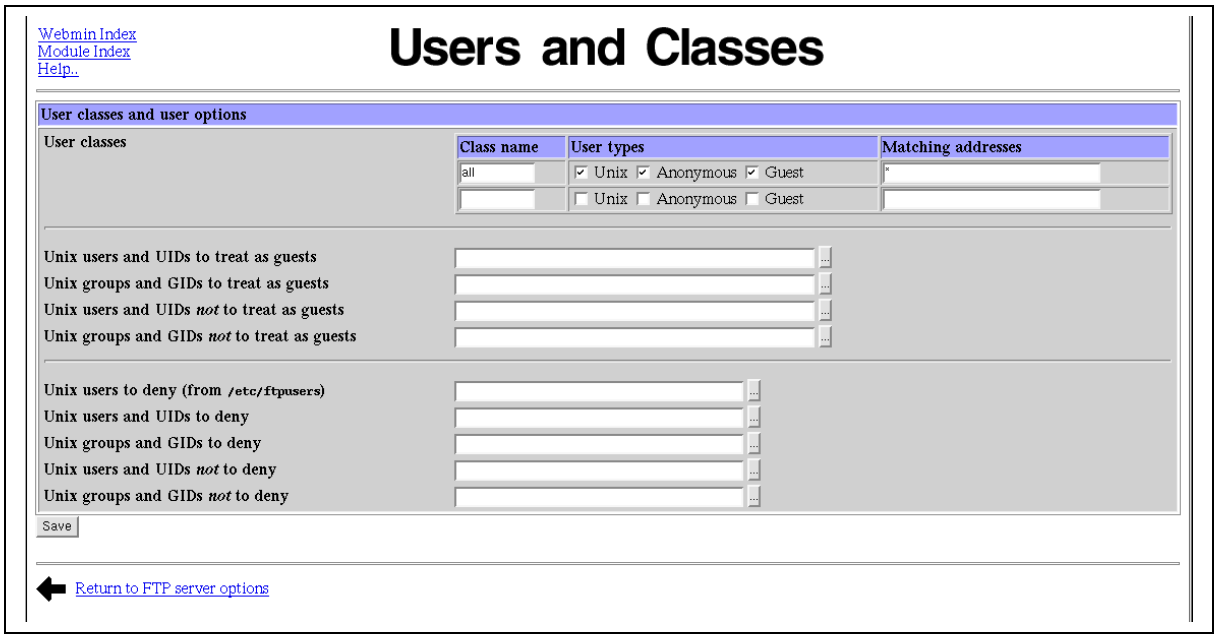


Nous accédons au menu principal de configuration du serveur qui nous rappelle en premier sa version (2.6) avant d'afficher neuf menu-icônes de configuration.



2. Users and Classes

Cette première section, au vu de son nom, permet de déclarer les classes d'utilisateurs du serveurs (**class**). Le terme d'utilisateurs réels est remplacé par celui d'Unix.

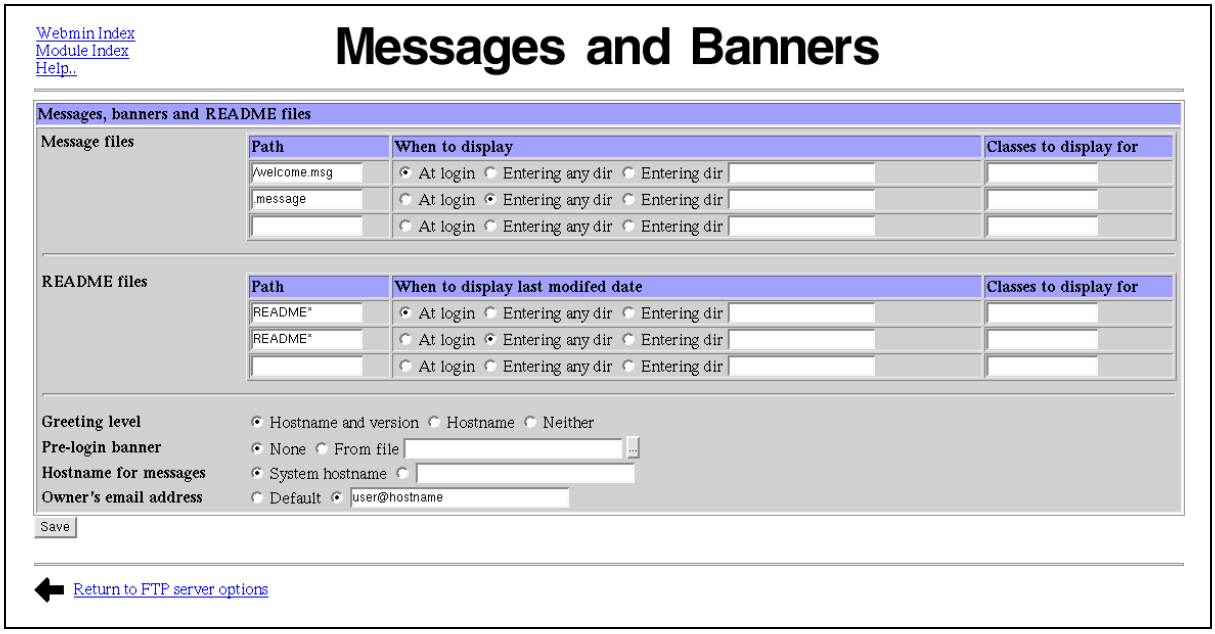


Les neuf champs textes qui suivent permettent à l'administrateur de délimiter avec précision les groupes d'utilisateurs et utilisateurs à traiter comme invités, ainsi que ceux à interdire (voir le fichier */etc/ftpusers*) : directives **unrestricted-uid**, **unrestricted-gid**, **restricted-uid**, **restricted-gid**, **deny-uid**, **deny-gid**, **allow-uid**, **allow-gid**.

3. Message and Banners

Cette deuxième section concerne la définition des différents messages et bannières envoyés en mode commande au client.

Dans un premier champ, il mentionne que le fichier *welcome.msg* contient le message d'accueil à la connexion ou *login* (**message**).
Dans un deuxième champ, le fichier *.message* est chargé d'afficher un avertissement à chaque parcours de répertoire (**message**).



Les fichiers README ont pour rôle de préciser le contenu du site dans sa globalité ou par répertoire parcouru (**readme**).

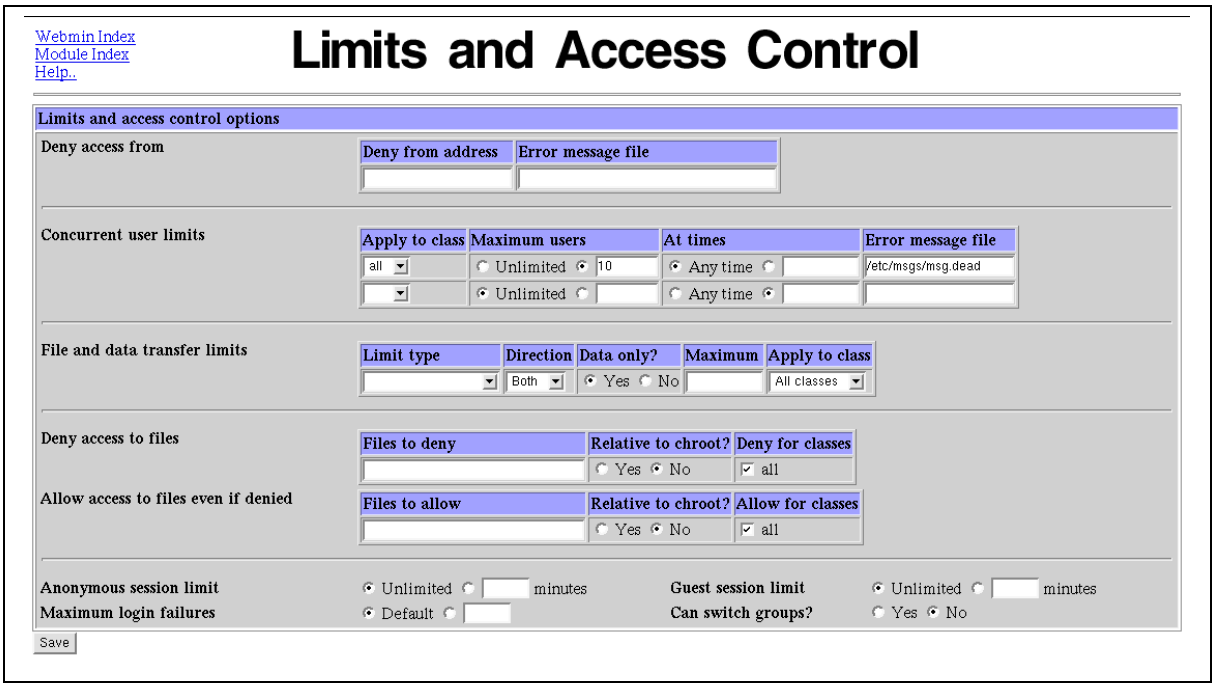
Les quatre derniers champs présentent quelques options supplémentaires :

- Le niveau d'informations renvoyées par le serveur sur ses identifiants (**greeting**),
- L'existence d'une bannière avant connexion (**banner**),
- Deux paramètres pouvant figurer dans les messages : nom d'hôte du serveur ou **hostname**, adresse de courriel de l'administrateur ou **email**.

4. Limits and Access Control

Cette troisième section a pour but :

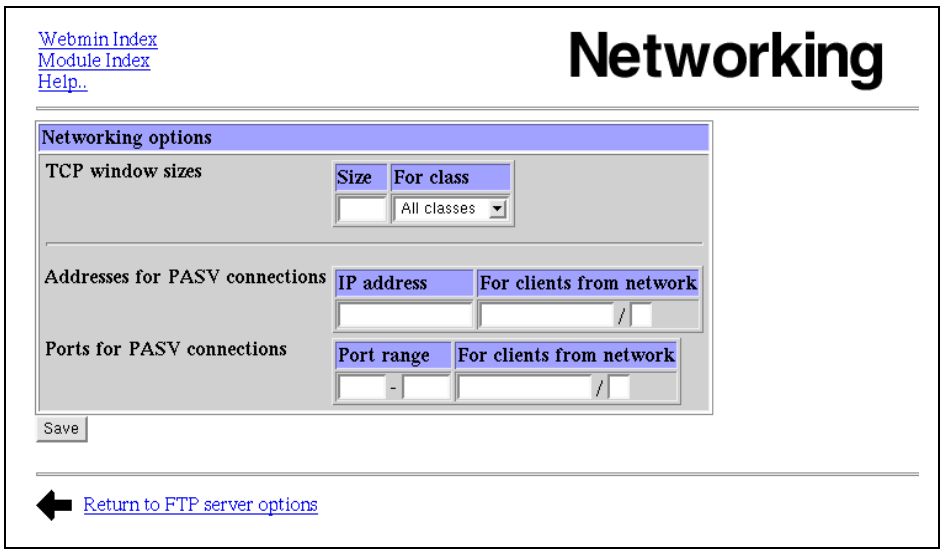
- D'interdire certains clients selon leur adresse IP ou DNS (**deny**), de les prévenir à travers le contenu d'un message ;
- De limiter le nombre de clients simultanés (**limit**) selon la classe d'appartenance, selon la période de connexion, de les prévenir à travers le contenu d'un message ;
- De limiter les transferts (**ul-dl-rate**), en précisant la méthode, le sens, le type des données (**data-limit**, **file-limit**), le nombre, la classe ;
- D'interdire l'accès à certains fichiers (**noretrieve**) ou d'autoriser malgré interdiction (**allow-retrieve**), selon la classe ;
- De fixer des délais de connexion pour les clients de type *anonymous* ou invités (**timeout connect**), d'arrêter un maximum de tentatives de connexion (**loginfails**), d'autoriser le changement de groupe au cours d'une session.



5. Networking

Cette quatrième section expose des paramètres réseau :

- Taille de fenêtre TCP ou bande passante (**tcpwindow**),
- Identifiants réseau des clients autorisés à travailler en mode passif : adresses IP (**passive address**), plages de numéros de port (**passive ports**) .

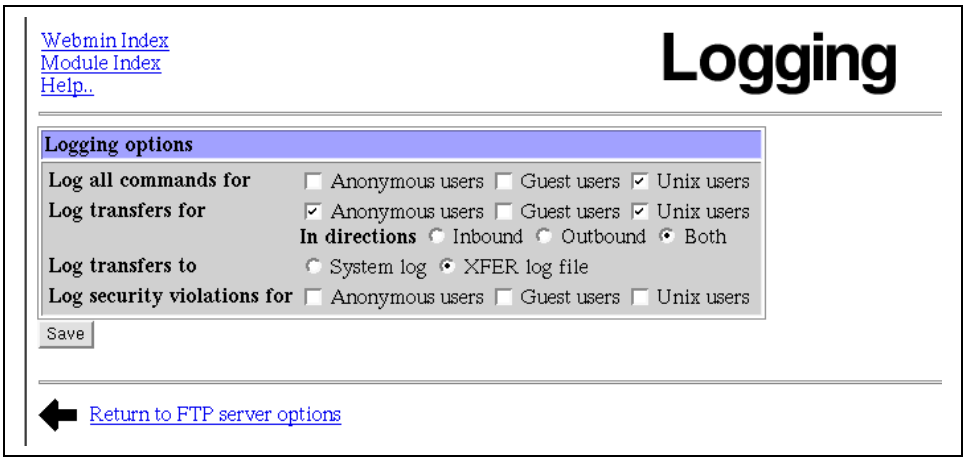


6. Logging

Cette cinquième section s'occupe plus particulièrement du contenu des fichiers journaux :

- Commandes des clients (**log commands**),
- Transferts des clients (**log transfers**),
- Sécurité des sessions (**log security**).

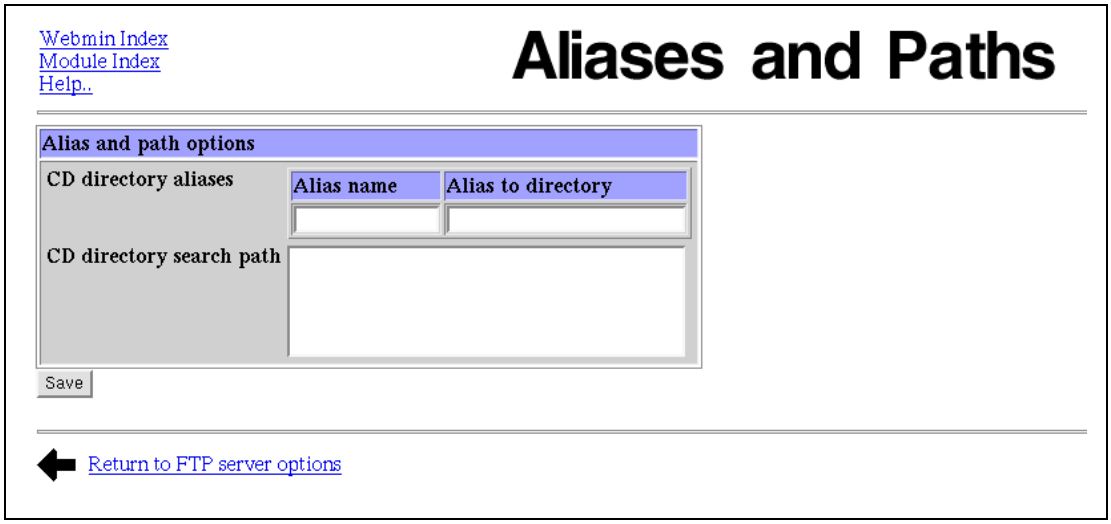
Elle permet en plus d'indiquer entre autres le nom du fichier journal, ainsi que le nom du gestionnaire de ces fichiers : processus système **syslog** ou non.



7. Aliases and Paths

Cette sixième section est optionnelle dans la mesure où elle permet de définir :

- des liens symboliques pour répertoires dans le premier champ (**alias**),
- des raccourcis dans le deuxième champ (**cdpath**).



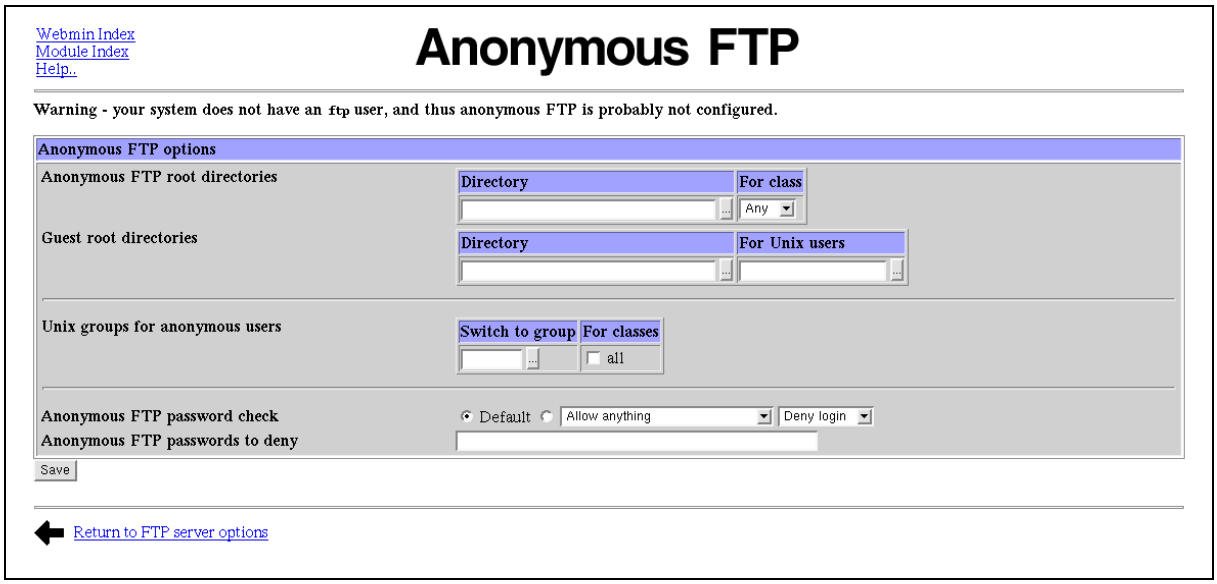
8. Anonymous FTP

Cette septième section est dédiée au site FTP public ou anonyme.

Elle permet de fixer :

- dans le premier champ la racine du site (**anonymous-root**),
- dans le premier champ de la ligne suivante celui des sites de comptes invités (**guest-root**).

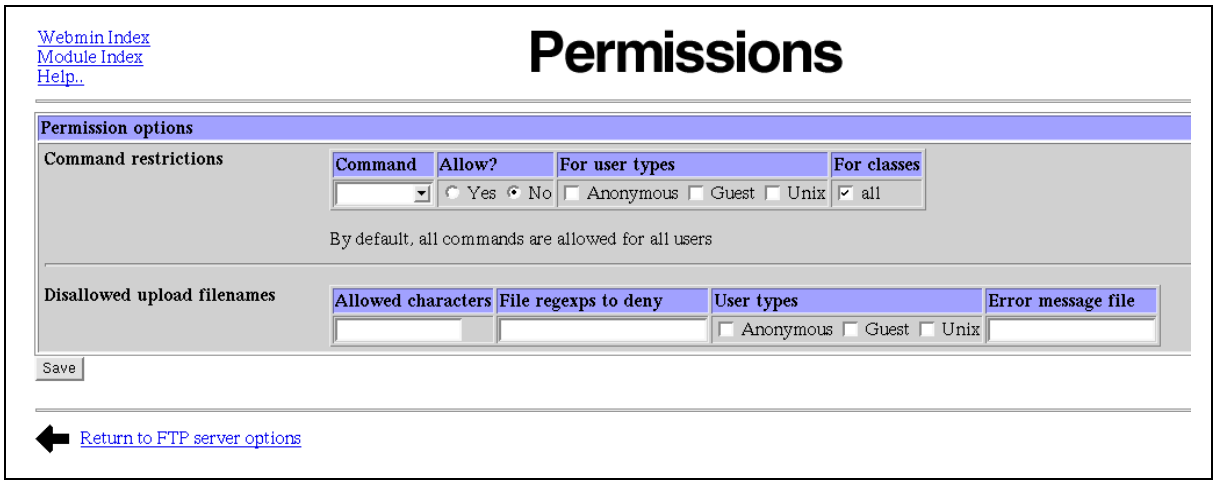
Le champ suivant permet d'associer des groupes système (ou réels) à des utilisateurs anonymes (**guestgroup**).
Les deux dernières lignes de paramétrage concernent la gestion du mot de passe (**password –check**, **deny -email**).



9. Permissions

Cette huitième section précise deux choses :

- les commandes autorisées pour les clients, surtout dans le cas d'un site de dépose (**chmod**, **delete**, **rename**, **overwrite**),
- les noms de fichiers autorisés à déposer (**path-filter**).

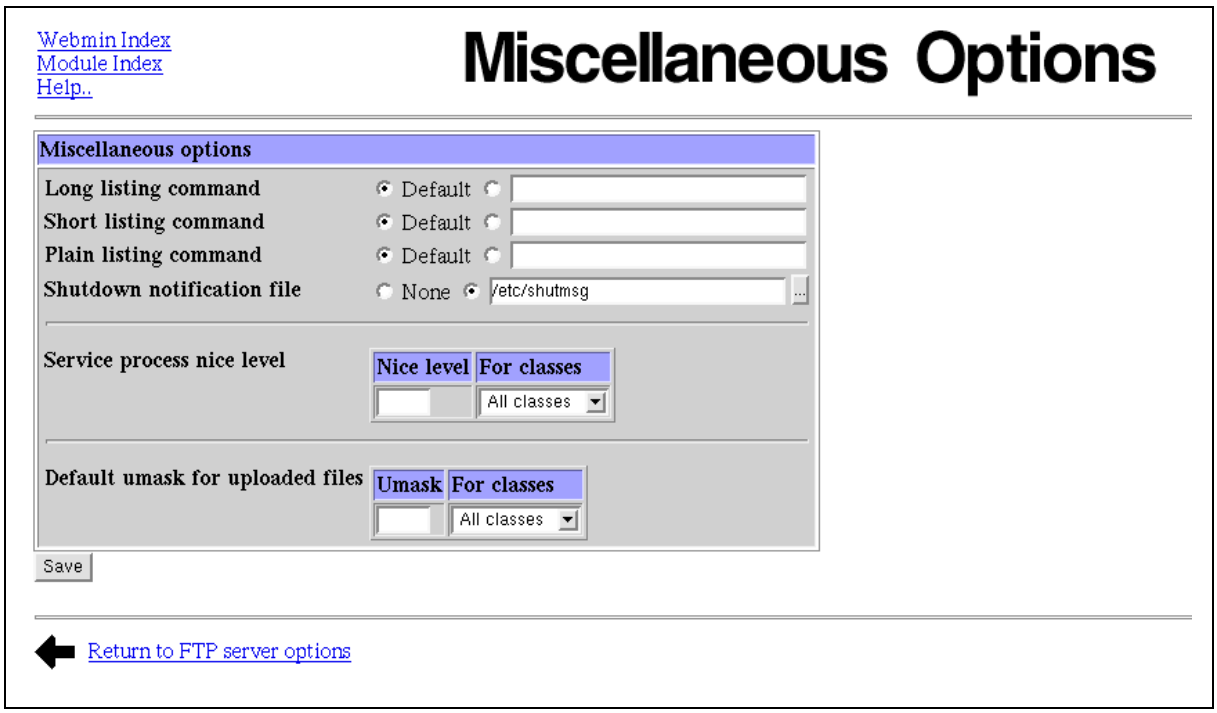


10. Miscellaneous Options

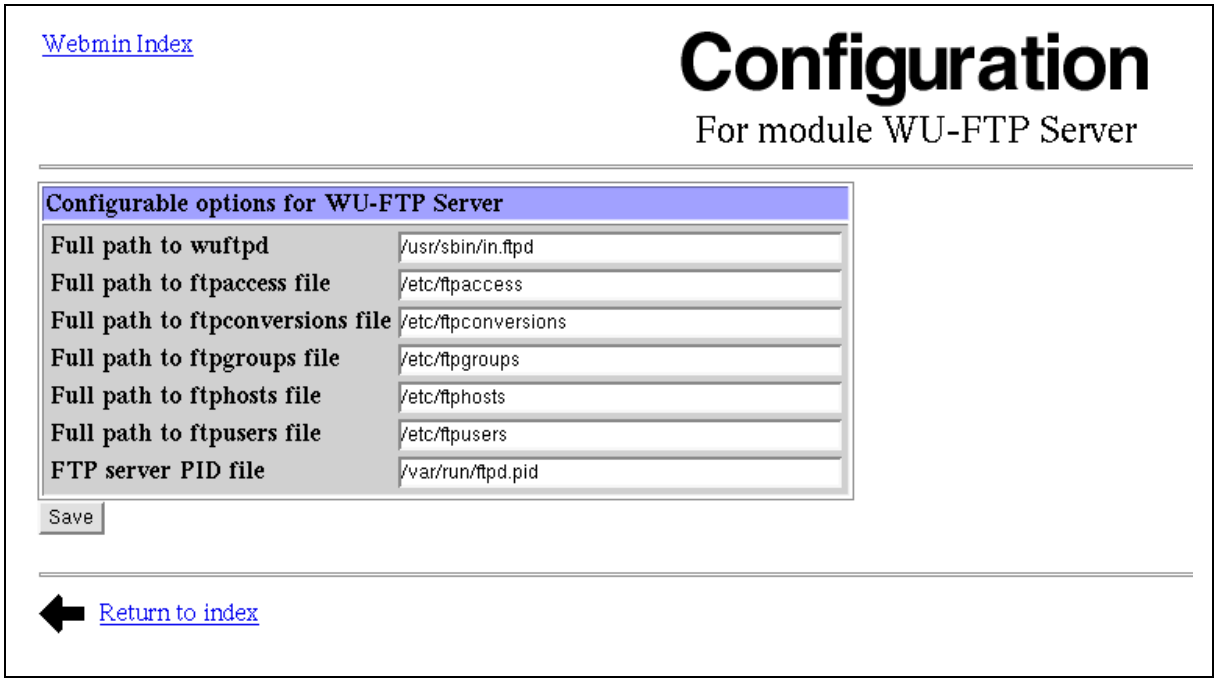
Cette neuvième section traite le reste des options non encore explicitées :

- Listing comportant peu (**lsshort**), beaucoup (**lslong**), ou toutes les informations (**lsplain**) ;

- Fichier de notification de fermeture du service (**shutdown**) ;
- Niveau de priorité du processus (**nice**) selon les classes d'utilisateurs ;
- Masque par défaut des fichiers déposés (**umask**) selon la classe d'utilisateurs.



11. Fichiers de Configuration du Logiciel



Dans le cas où l'administrateur du serveur n'a pas choisi les localisations standards pour les fichiers de configuration du serveur, il peut l'indiquer de manière interne au logiciel Webmin en cliquant sur le lien hypertexte *Module Index* situé en haut à gauche de chaque page (voir copie d'écran précédente).

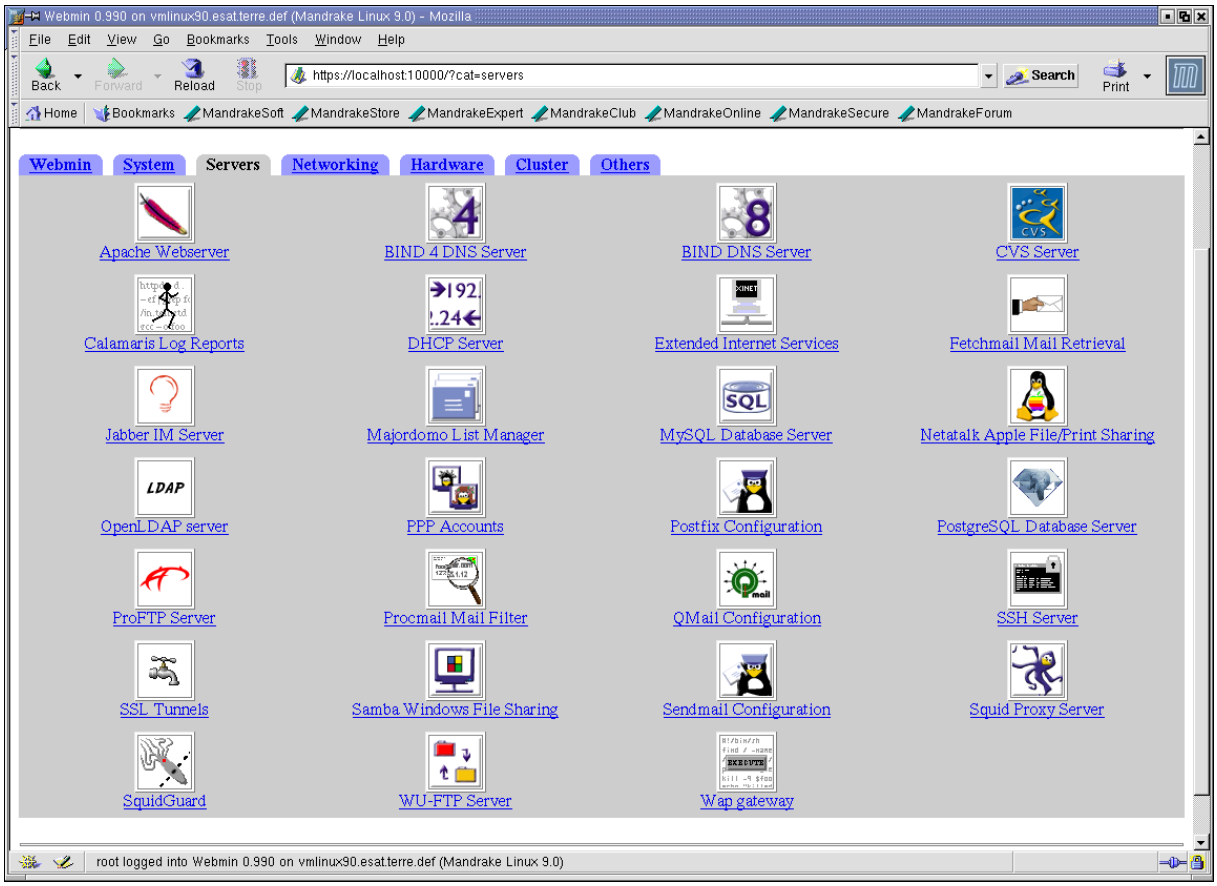
Annexe 3 : Administration distante Webmin de ProFTP

Le document présent suppose que le programme Webmin, issu du monde des logiciels libres, ait été installé sur la station hébergeant le serveur FTP, ainsi que le serveur ProFTP lui-même.

1. Menu général

Il est rappelé que ce logiciel permet l'administration à distance aussi bien du système que celui des services Internet, le tout via un interface Web.
Vu la gravité des sujets traités, son accès se fait via une connexion sécurisée (https) sur un numéro de port spécifique (10.000).

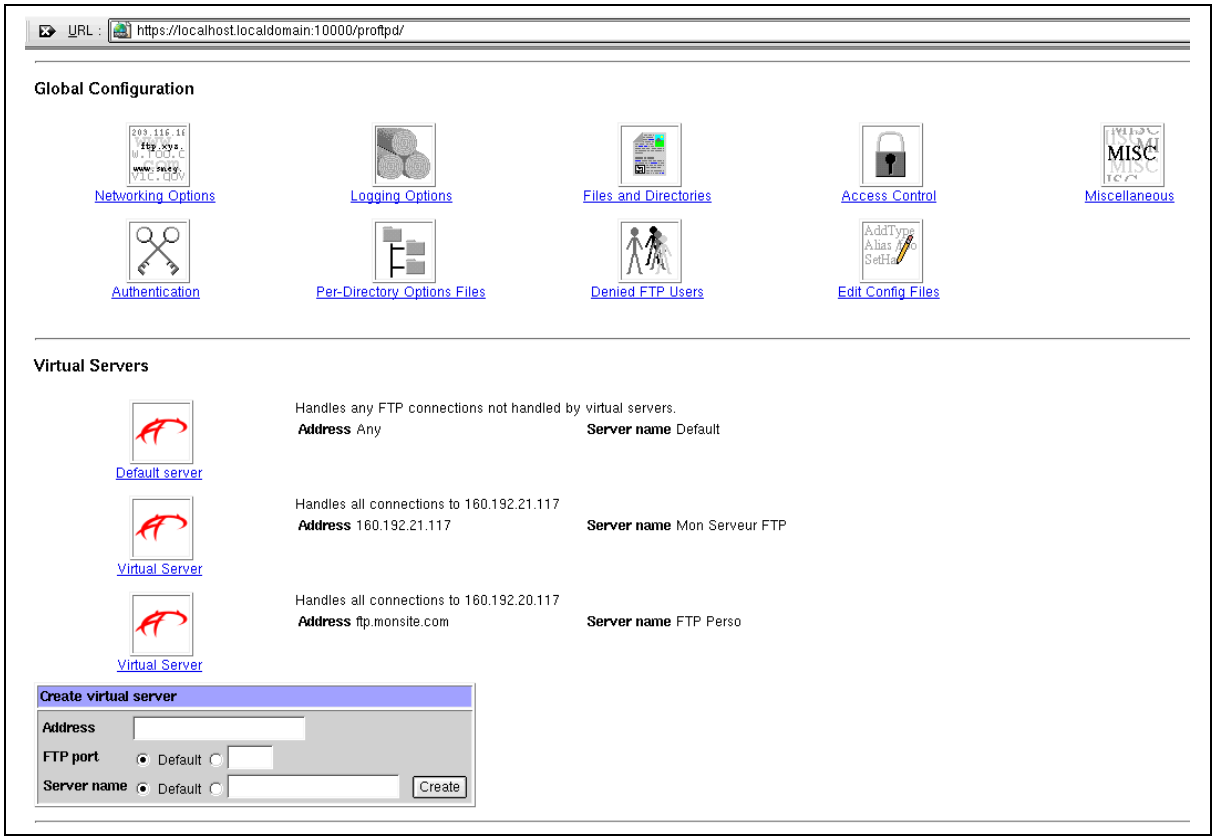
Une fois l'onglet *Serveurs* sélectionné, nous obtenons l'écran suivant.



Dans la première colonne à la cinquième ligne, nous pouvons activer le serveur *Proftpd* (copie d'écran suivante) pour accéder au menu principal de configuration du serveur.

Cette page Web se découpe en deux parties : une concernant la configuration globale du serveur, une autre sur les serveurs virtuels dont le serveur principal. La suite du document reprend l'organisation de cette page.

Une fois les changements opérés, un bouton *Apply Changes*, situé en bas de la page mais non visible dans la copie d'écran, permet de faire prendre en compte les modifications par le serveur.



2. Configuration globale

La partie qui décrit la configuration globale comprend neuf sections accessibles via les liens hypertextes et hypermédia.

11.1 Networking Options

Le premier menu décrit à travers les options de configuration du réseau, les principales directives de fonctionnement du serveur, en commençant par son mode de travail (**ServerType**) :

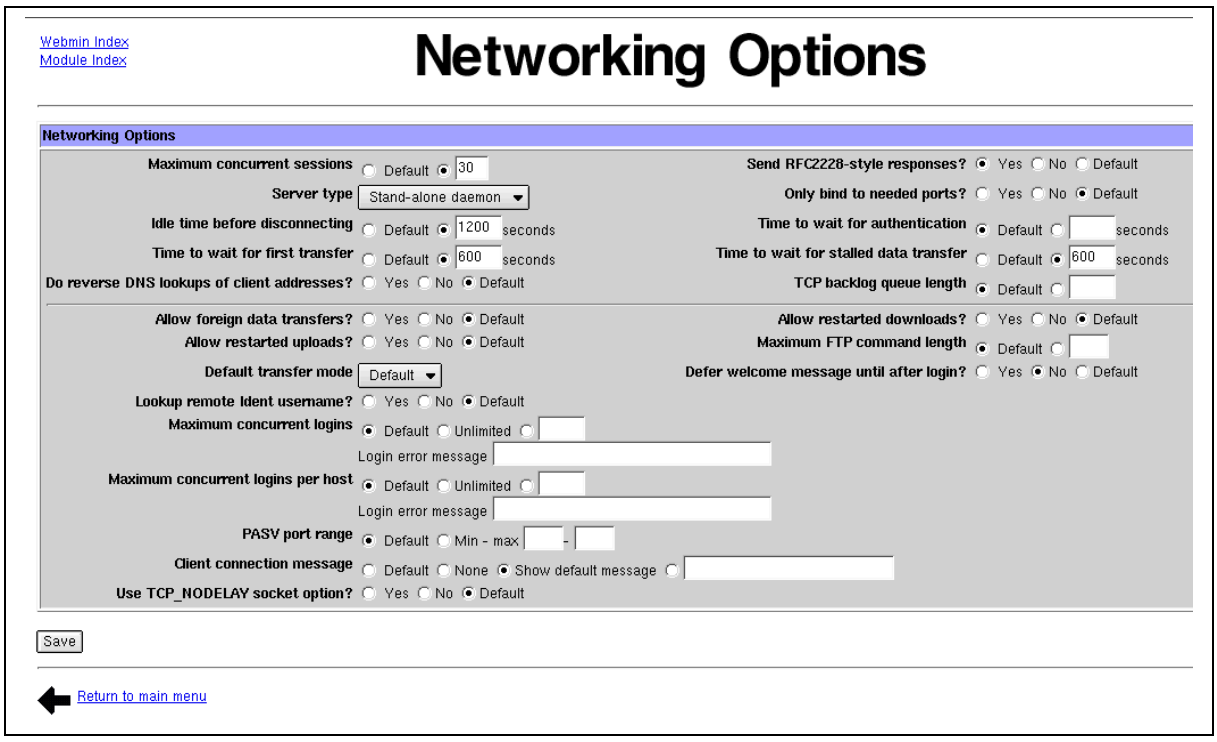
a) Respectivement le délai de connexion inactive (**TimeOutIdle**), le délai sans transfert (**TimeOutNoTransfer**), le délai d'authentification (**TimeOutLogin)** et le délai des téléchargements bloqués (**TimeOutStalled**) ;

b) Les paramètres sur l'affichage (**MultilineRFC2228**) ou les informations renvoyées par le serveur (**DeferWelcome**) sur lui-même ;

c) Le nombre maximum de clients (**MaxInstances),** le nombre maximum depuis une station donnée (**MaxClientsPerHost**), le nombre d'identités simultanés du client quel que soit l'endroit depuis lequel il opère (**MaxClientsPerUser**) ;

d) Le mode de transfert par défaut côté serveur (**DefaultTransferMode**) : ASC ou BIN ;

e) L'archivage des adresses DNS des clients au lieu de leurs adresses IP (**UseReverseDNS**), la recherche des noms d'utilisateurs (**IdentLookups**) ;
f) L'autorisation des transferts d'un client en mode passif depuis un autre ordinateur que celui déclaré (**AllowForeignAddress**) ;
g) L'autorisation de la reprise de transfert de dépose (**AllowStoreRestart**), et celle de téléchargement (**AllowRetrieveRestart**) ;
h) La définition de la plage de numéros de port en mode passif (**PassivePorts**) ;
i) Le choix du message affiché à la connexion (**DisplayLogin**) ;
j) Par défaut, la suppression du délai d'attente au niveau des sockets TCP (**tcpNoDelay**);
k) L'assignation des numéros de port nécessaires (**SocketBindTight**) ;
l) La modification si besoin de la longueur de réponse d'une trame TCP en mode autonome (**tcpBackLog**) ;
m) Le choix si besoin de la longueur des commandes du client (**CommandBufferSize**).

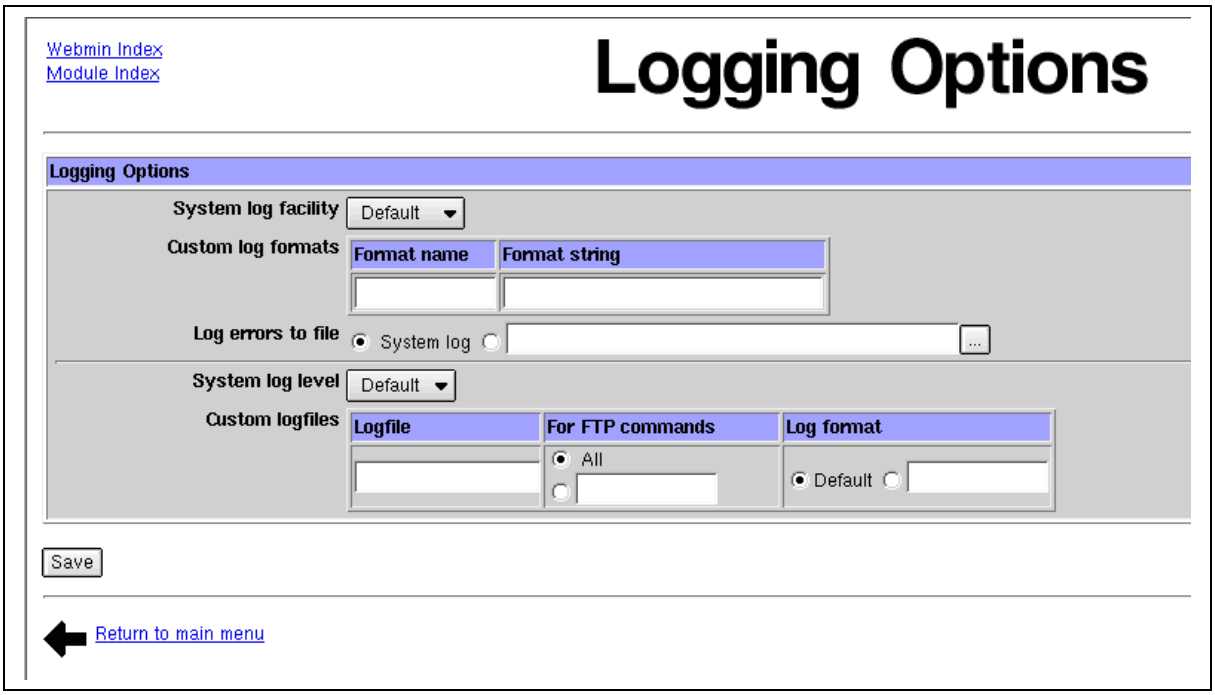


11.2 Logging Options

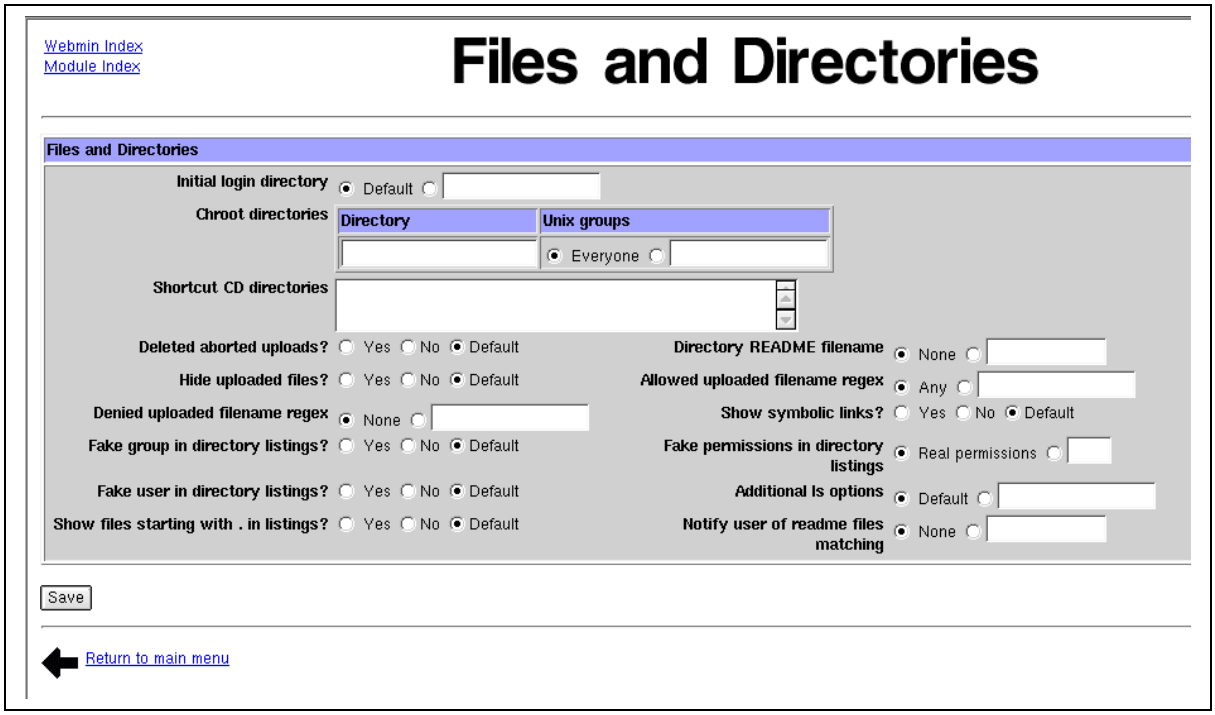
Le deuxième menu ci-avant décrit la gestion des fichiers journaux :

a) Le compte utilisateur vers lequel seront transmis ces informations (**SyslogFacility**)
b) Le nom du format personnalisé (**LogFormat**) suivi de la chaîne de paramètres constitutifs ;
c) La redirection des enregistrements vers les fichiers journaux systèmes propres au processus *syslog* (**SystemLog**), sinon le nom du fichier journal spécifique si besoin (**ExtendedLog**) ;
d) Le choix d'historiser toutes les commandes FTP ou non (ALL ou WRITE ou …);
e) Le niveau d'informations (liste *log level*) à historiser (**SyslogLevel**) :

- Les informations critiques, les informations urgentes, celles d'alerte,
- Les informations d'erreur, de débogage,
- Les messages d'avertissement de notifications et d'informations pures.



11.3 Files and Directories



Le troisième menu décrit :

a) Le répertoire initial de connexion (**DefaultRoot**), le déroutage de ces derniers (**DefaultChdir**) selon le groupe concerné, les raccourcis des chemins de connexion (**CDPath**) ;
b) Les options autorisées pour la commande *ls* (**LsDefaultOptions**) ;
c) Les opérations permises sur les répertoires du site : cacher les fichiers déposés (**HiddenStor(es)**), suivre les liens symboliques (**ShowSymLinks**) ;
d) Le fichier affiché lors du passage dans un répertoire (**DisplayFirstChdir**) ;
e) Le choix des expressions régulières de commandes autorisés (**PathAllowFilter**) ou interdites (**PathDenyFilter**) pour la dépose.

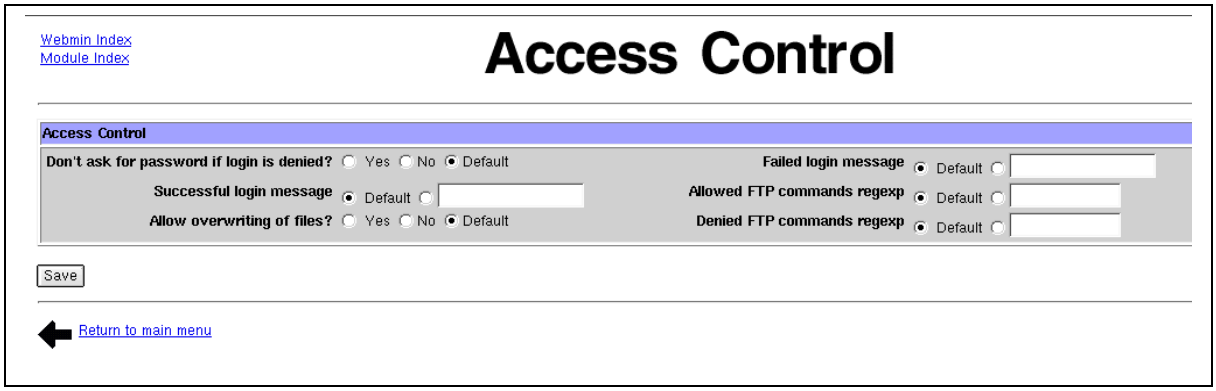
L'administrateur peut décider :

- de cacher aux clients le nom des vrais propriétaires (**DirFakeUser**) et groupes d'appartenance (**DirFakeGroup**) des répertoires et fichiers proposés ainsi que leurs droits (**DirFakeMode**) ;
- de notifier à l'utilisateur la modification des fichiers README (**DisplayReadMe**) ;
- de cacher les fichiers dont le nom commence par un point (**ShowDotFiles**) ;
- de détruire les fichiers dont le transfert a été interrompu (**DeleteAbortedStores**).

11.4 Access Control

Le quatrième menu permet :

- D'ôter la demande de mot de passe lors de l'authentification en cas de nom d'utilisateur interdit (**LoginPasswordPrompt**);
- de fixer le texte personnalisé du message renvoyé en cas d'échec d'authentification (**AccessDenyMsg**), comme celui envoyé en cas de réussite (**AccessGrantMsg**) ;
- de décider de l'écrasement des fichiers existants (**AllowOverwrite**) ;
- du choix des commandes *regexp* autorisées ou interdites (commandes TCL de remplacement d'expression régulière par une chaîne de caractères).

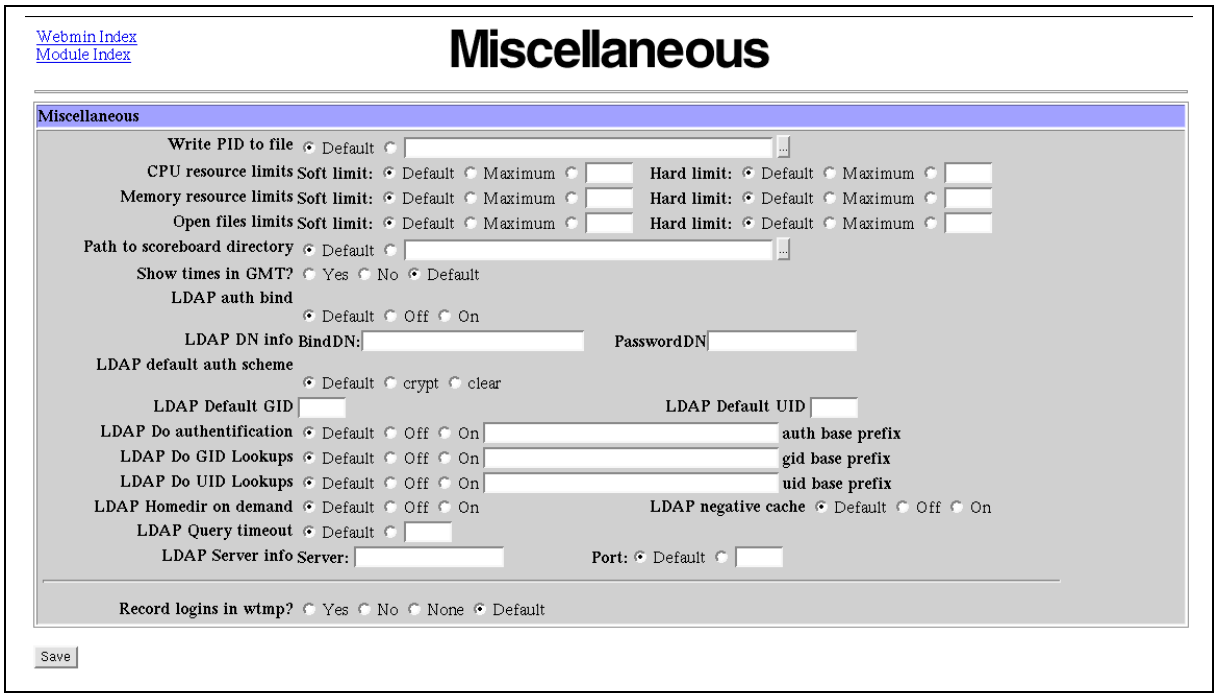


11.5 Miscellaneous

Le cinquième menu propose des paramètres optionnels concernant :

- des ressources système ;

- stocker l'identifiant des processus serveurs dans un fichier (**PidFile**),
- limiter les ressources en CPU (**RLimitCPU**) ou en mémoire (**RlimitMemory**) sur un petit serveur, ou en nombre de fichiers ouverts (**RLimitOpenFiles**), au niveau logique (soft) comme au niveau physique (hard),
  - le chemin du répertoire d'affichage (**ScoreBoardFile**) ;
  - le fuseau horaire d'affichage des dates (**TimesGMT**) ;
- des paramètres communs (au nombre de 21) avec un serveur LDAP (**LDAP…**).

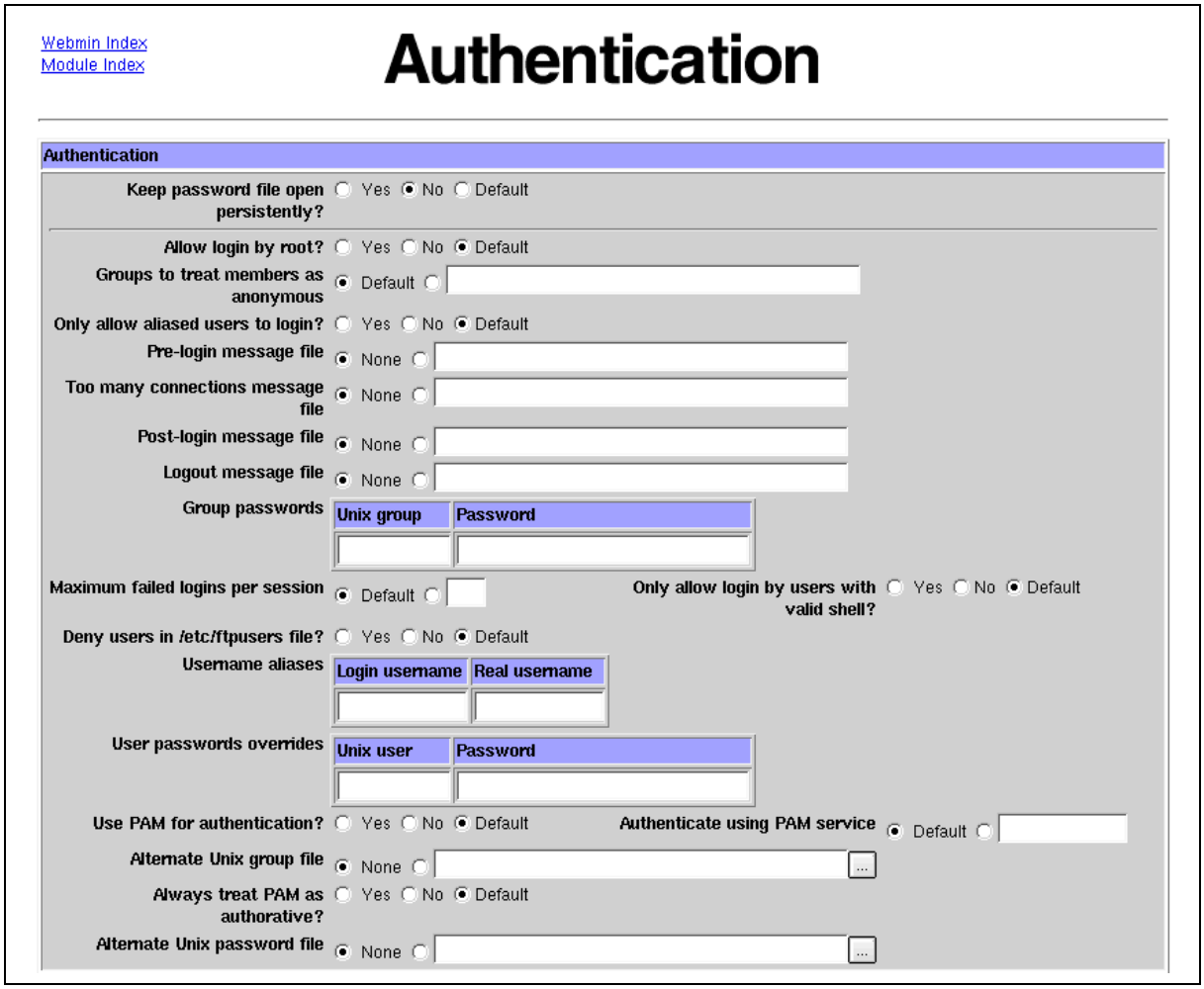


11.6 Authentification

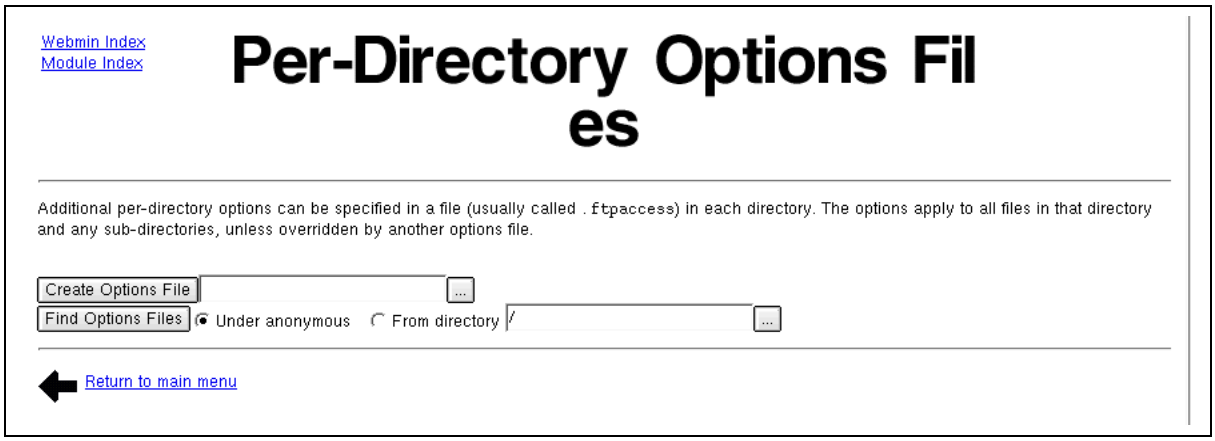
Le sixième menu traite particulièrement des problèmes d'authentification :

- Utiliser le fichier de déclaration des utilisateurs du système (**PersistentPasswd**) pour valider les connexions des clients ;
- Autoriser l'identité de l'administrateur ou *root* (**RootLogin**) ;
- Autoriser des groupes d'utilisateurs comme utilisateur *anonyme* (**AnonymousGroup**);
- Définir les alias d'utilisateurs (**UserAlias**) et uniquement autoriser ces derniers (**AuthAliasOnly**) ;
- Afficher un message avant connexion (**DisplayLogin**), afficher un message après déconnexion (**DisplayQuit**) ;
- Afficher un message si trop de clients sont connectés simultanément (**LeechRatioMsg**) ;
- Afficher un message après connexion (**AccessGrantMsg**) ;
- Gérer les mots de passe de groupes (**GroupPassword**) ;
- Autoriser un maximum d'échecs de connexion (**MaxLoginAttempts**) ;
- Définir les interpréteurs de commandes autorisés (**RequireValidShell**) ;

- Interdire les utilisateurs (par défaut) décrits dans le fichier */etc/ftpusers* (**UseFtpUsers**);
- Créer des couplets utilisateur / mot de passe (**UserPassword**) ;
- Utiliser le service système complémentaire PAM de Linux (**AuthPAM),** définir le nom du service d'authentification (**AuthPAMConfig**), indiquer si le service PAM prend la main sur les autres systèmes d'authentification (**AuthPAMAuthoritative**) ;
- Changer de fichier de déclaration des groupes (**AuthGroupFile**) ou de mots de passe (**AuthUserFile**).



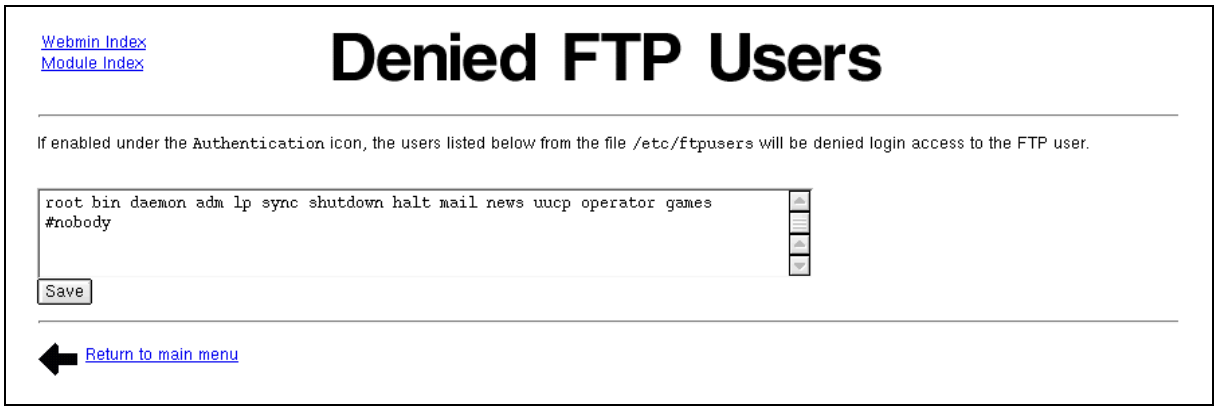
11.7 Per-Directory Options Files



Le septième menu traite des opérations permises sur les fichiers. Il permet de définir par répertoire un fichier caché *.ftpaccess* qui décrit les permissions de ce répertoire.

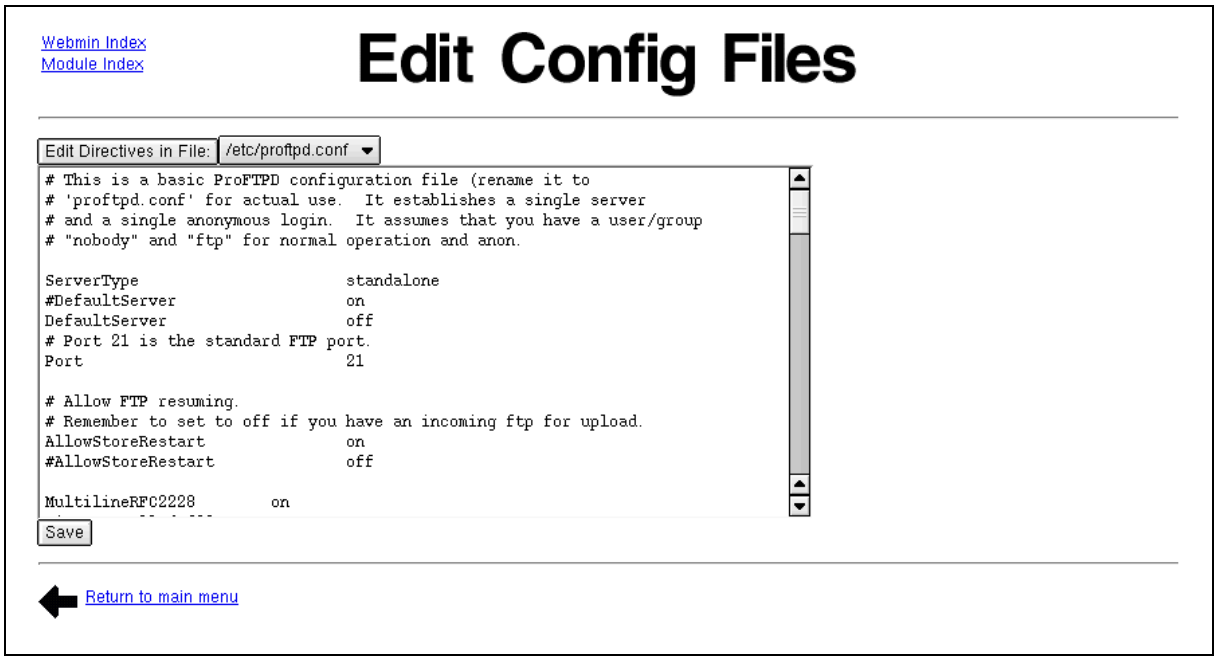
11.8 Denied FTP Users

Le huitième menu traite des utilisateurs système interdits.



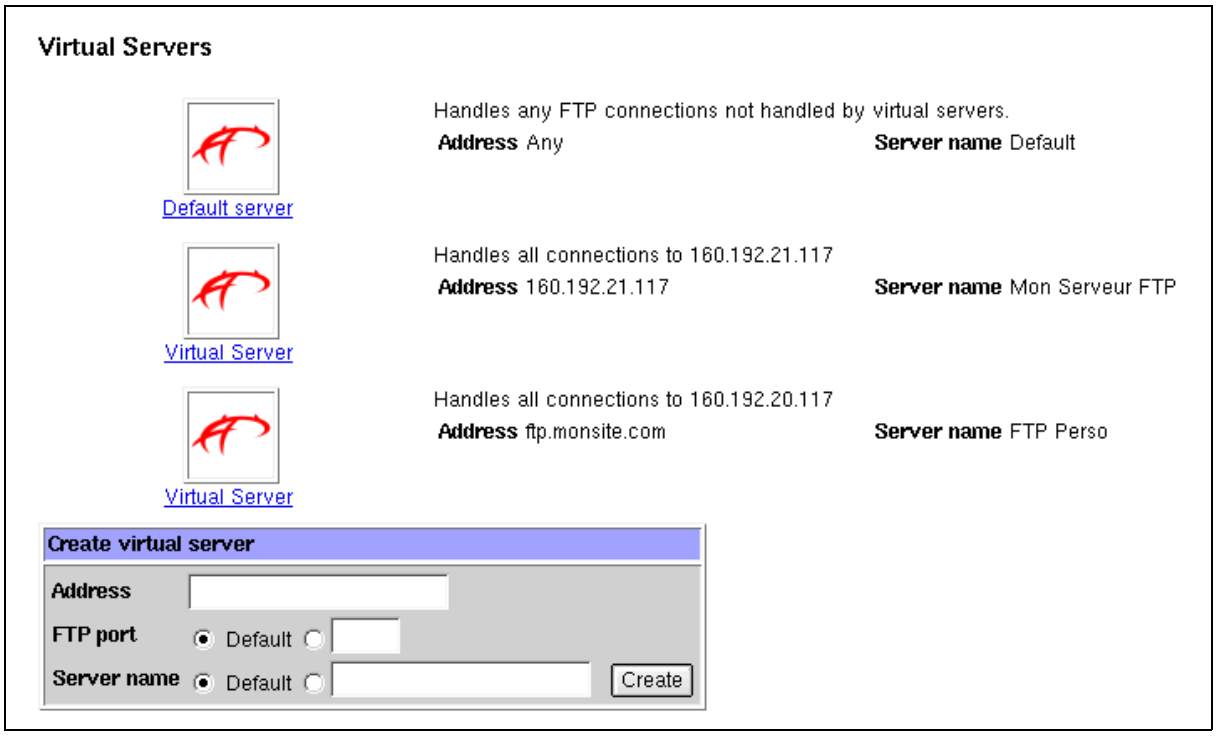
11.9 Edit Config Files

Le neuvième menu affiche le contenu du principal fichier de configuration.



3. Serveurs virtuels

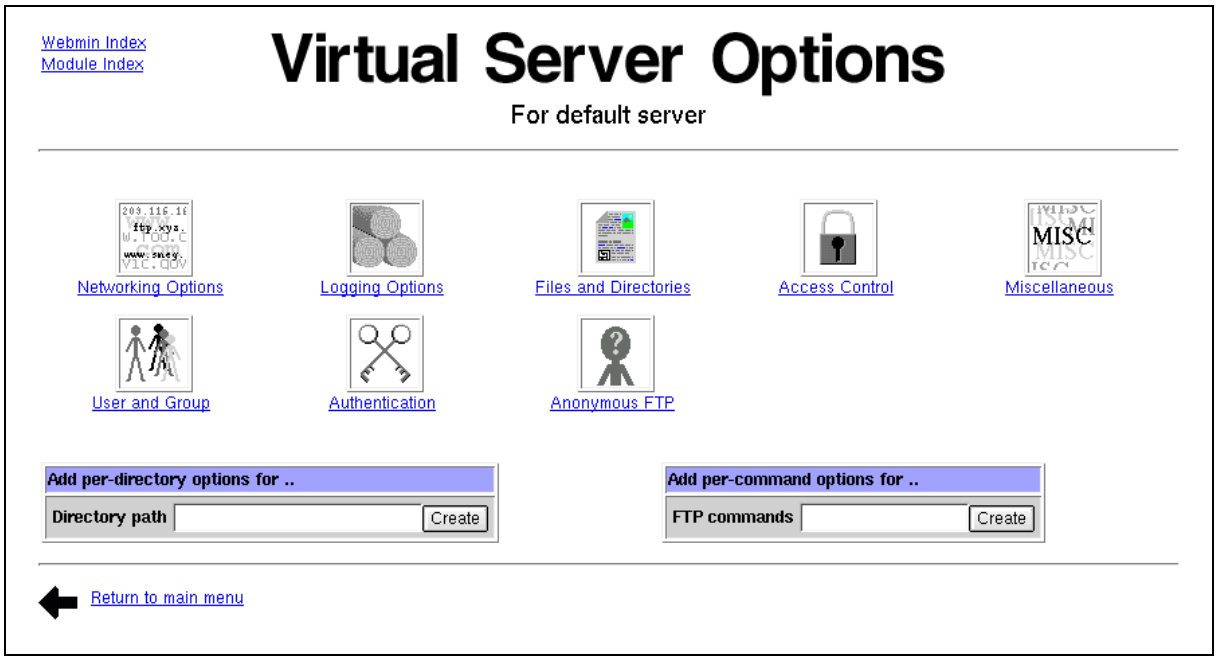
Cette copie d'écran issu du menu principal de configuration permet d'accéder aux sous-menus de configuration du serveur par défaut, puis de ceux d'un serveur virtuel par défaut.
Si l'administrateur veut configurer un deuxième serveur virtuel, le menu-icône est déjà disponible.



Dans la suite du document, nous passerons en revue d'abord les paramètres du serveur par défaut, puis ceux d'un serveur virtuel.

11.10 Serveur par défaut

L'activation du lien sur le site (ou serveur) par défaut offre l'accès à une page principale offrant des menus similaires au menu de configuration principal du serveur.



Les menus icônes *Per-Directory Options Files*, *Denied FTP Files* et *Edit Config Files* sont absents alors que de nouveaux menus apparaissent : **User and Group** et **Anonymous FTP**.

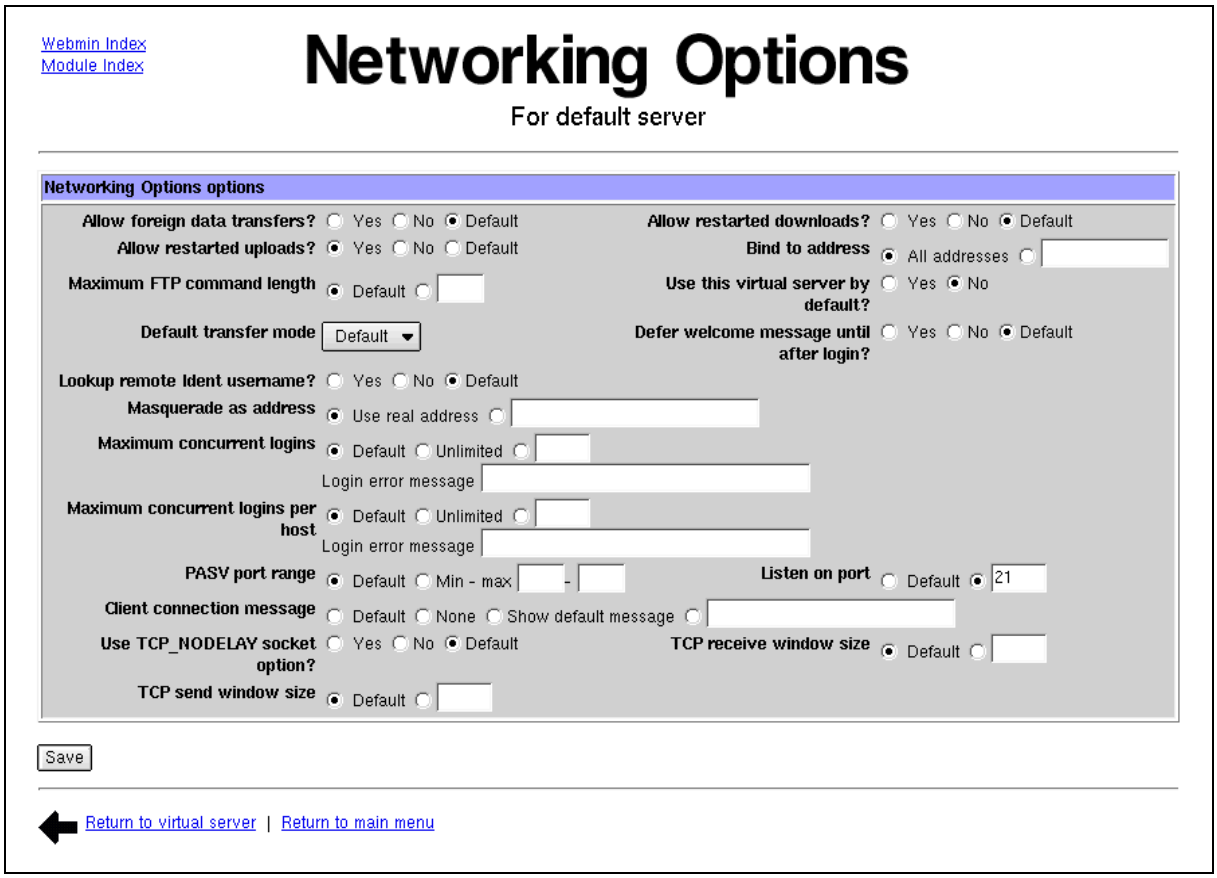
Sous les menus icônes apparaissent deux champs :

- celui des répertoires protégés (**Per-Directory Options**),
- celui des commandes FTP autorisées sous ce répertoire (**Per-Command Options**).

11.10.1. Networking Options

Ce premier menu offre une multitude d'options appliquées au niveau du serveur :

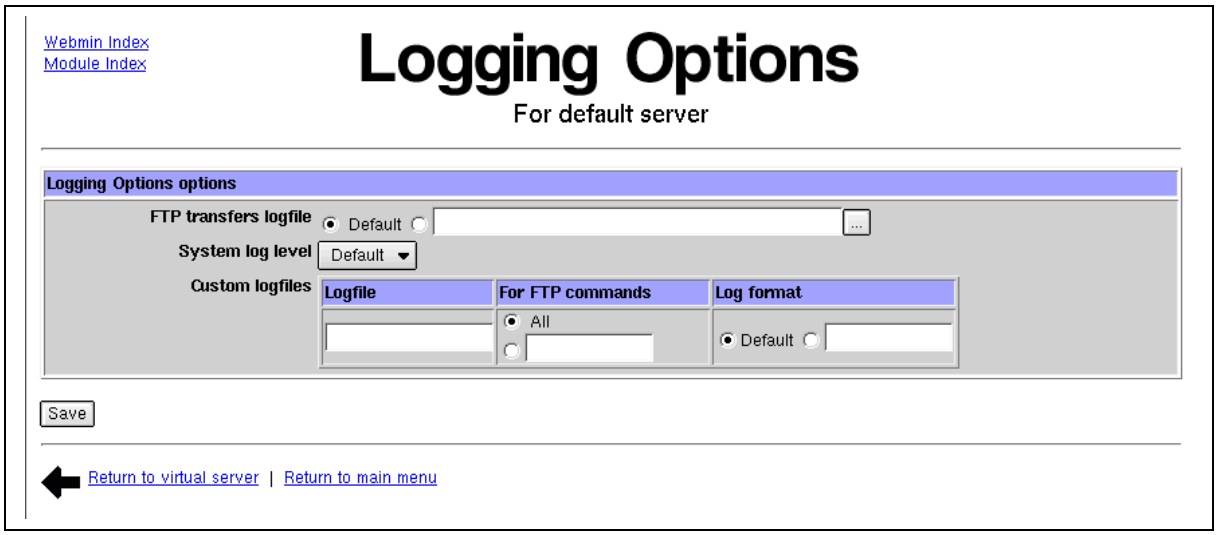
- Autoriser les transferts d'un client depuis un hôte distant autre que l'hôte habituel (**AllowForeignAdress)** ;
- Autoriser les reprises de transferts en téléchargements (**AllowRetrieveRestart**) ou en déposes (**AllowStoreRestart**) ;
- Assigner une adresse IP au site (**Bind**), et définir son numéro de port TCP (**Port**) ;
- Désigner ce site comme serveur par défaut (**DefaultServer**) ;
- Définir le message affiché au client à la connexion (**DisplayLogin**), afficher un message minimum sur le serveur (**DeferWelcome**) ;
- Définir la longueur maximale des commandes (**CommandBufferSize**), le mode de transfert par défaut (**DefaultTransferMode**) ;
- Surveiller le nom d'utilisateur distant (**IdentLookups**) ;
- Masquer l'adresse du serveur (**MasqueradeAddress**) ;
- Limiter le nombre de clients simultanés (**MaxClients**) et définir le message envoyé en cas de seuil atteint (**LeechRatioMsg**) ;



- Limiter le nombre de clients simultanés par hôte distant (**MaxClientsPerHost**) ainsi que le message d'erreur à envoyer si le seuil est atteint (**DisplayGoAway**) ;
- Définir la plage de numéros de port TCP utilisés en mode passif (**PassivePorts**);
- Utiliser des *sockets* TCP sans délai d'attente (**tcpNoDelay**) ;
- Fixer une bande passante en émission (**tcpSendWindow**) ou en réception (**tcpReceiveWindow**) ;

11.10.2. Logging Options

Le deuxième menu décrit la gestion des fichiers journaux avec des options déjà rencontrées dans le menu principal (paragraphe 1.2) :



a) Le nom du fichier journal des transferts personnalisé si besoin (**TransferLog**) ;
b) Le niveau d'importance des évènements à enregistrer (**SyslogLevel**) : du niveau informatif au niveau critique ;
c) Le nom du fichier journal spécifique si besoin (**ExtendedLog**) suivi du choix d'historiser toutes les commandes FTP ou non (ALL ou WRITE ou …) ;
d) Le choix d'un format des enregistrements spécifiques (**LogFormat**).

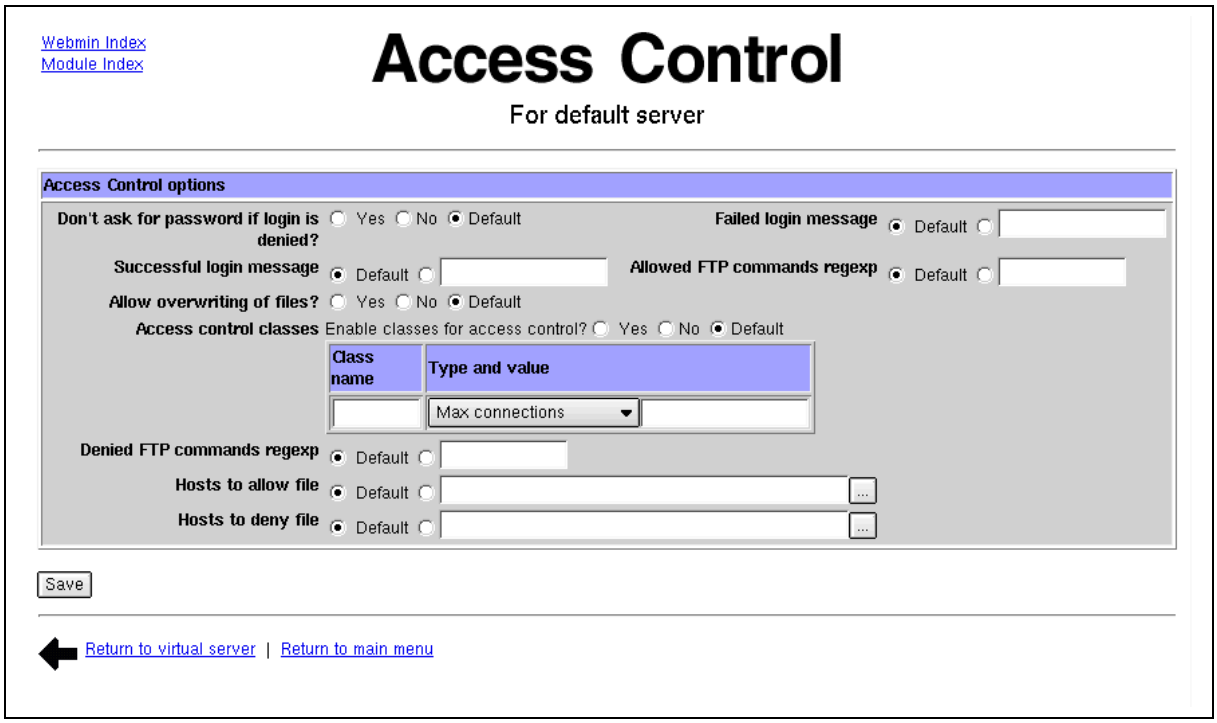
11.10.3. File and Directories

Ce troisième menu offre les mêmes options que le menu équivalent du contexte global de configuration (paragraphe 1.3).

11.10.4. Access Control

Le quatrième menu, outre les options déjà rencontrées dans le contexte global de configuration (paragraphe 1.4), permet :

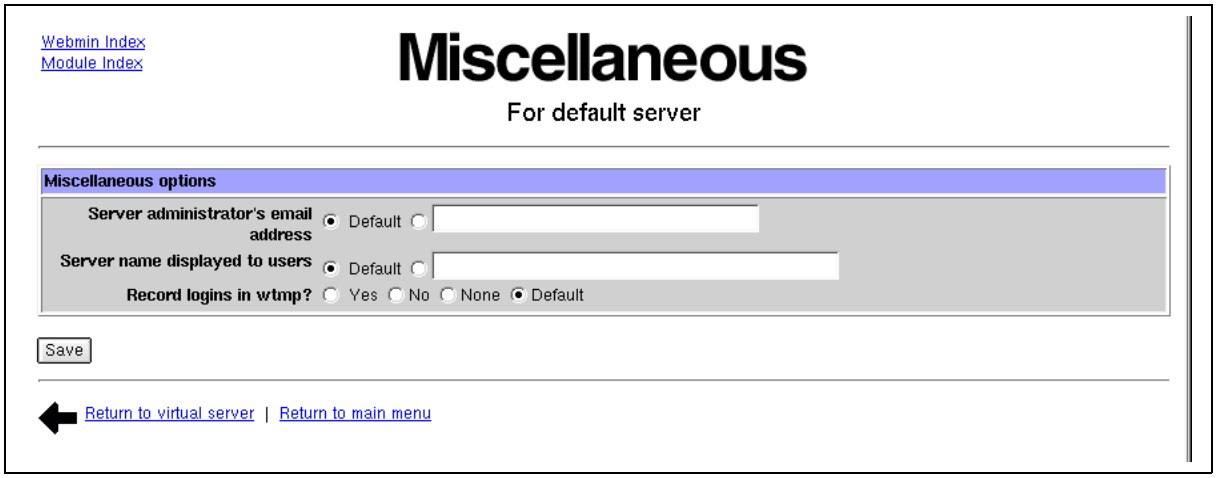
- de particulariser les accès selon la classe d'utilisateurs concernée (**Class**) ;
- de préciser les commandes *regexp* ;
- d'utiliser les fichiers système du réseau énumérant les hôtes (clients) autorisés (*/etc/hosts.allow*) ou interdits (*/etc/hosts.deny*).



11.10.5. Miscellaneous Options

Le cinquième menu offre la possibilité :

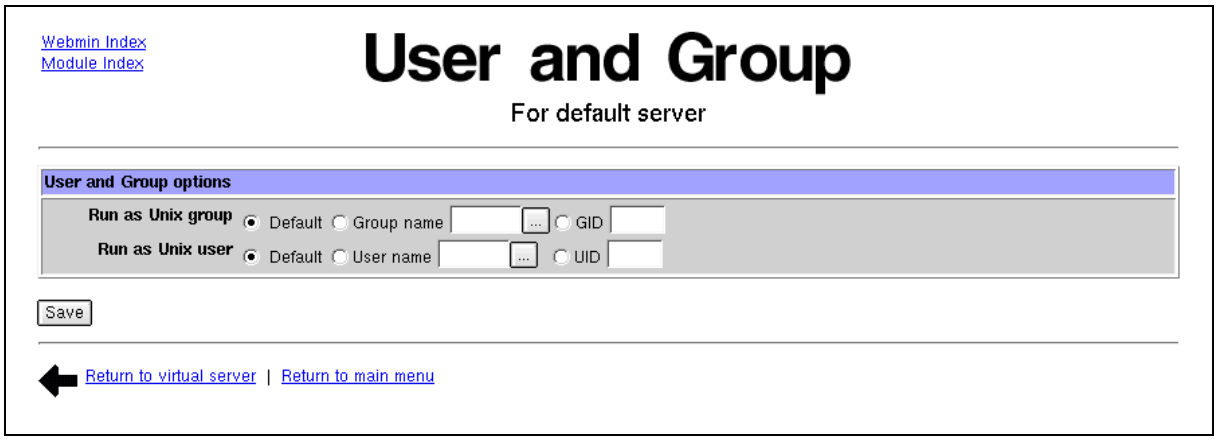
- de définir l'adresse de courriel de l'administrateur (**ServerAdmin**) ;
- d'afficher le nom du serveur aux utilisateurs connectés (**ServerName**) ;
- d'enregistrer les connexions dans un fichier système (/var/log/wtmp) de Linux (**WtmpLog**).



11.10.6. User and Group

Ce sixième menu précise si besoin est :

- Les groupes d'utilisateurs invités par leur nom et leur GID ;
- Les utilisateurs invités par leur nom et leur UID.



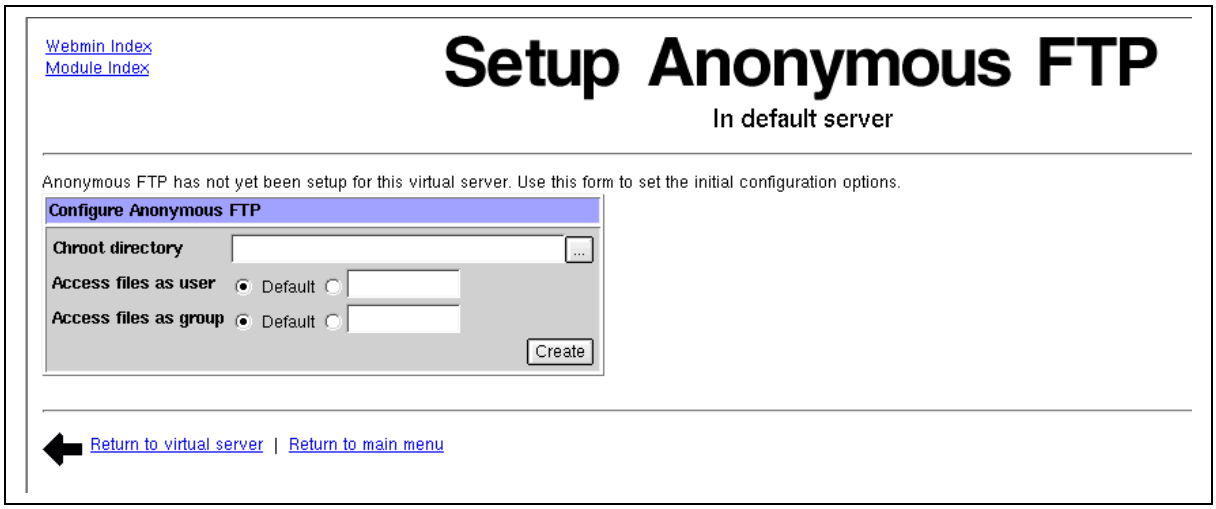
11.10.7. Authentification

Le septième menu traite de l'authentification au niveau de ce site. Il reprend à une exception près (directive **PersistentPasswd**) toutes les options déjà décrites dans le contexte de configuration globale (paragraphe 1.5).

11.10.8. Setup Anonymous FTP

Le huitième menu décrit la configuration d'un site FTP public (ou anonyme) et rappelle la possibilité de choisir un répertoire autre que celui par défaut (chroot ou **DefaultRoot**).

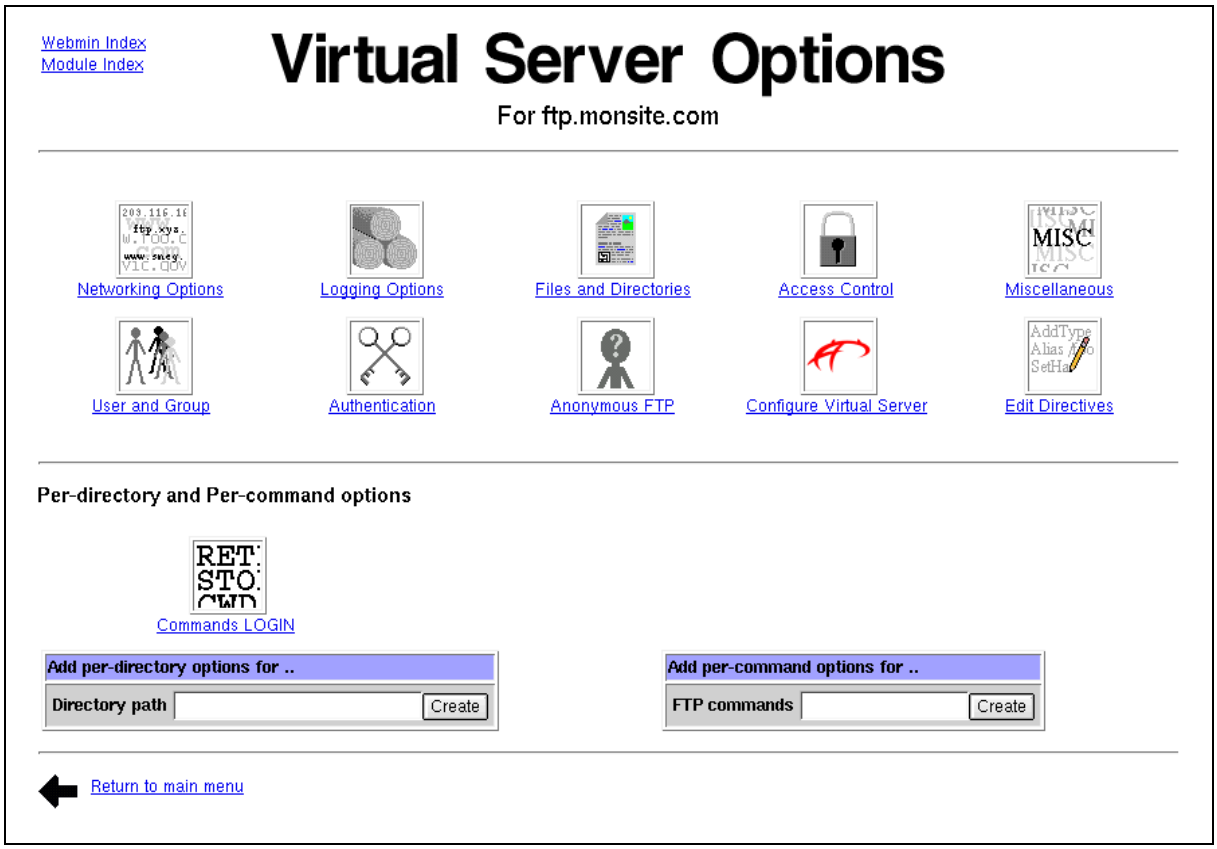
Il permet aussi de désigner les fichiers de permissions d'accès des utilisateurs (**AuthUserFile**) et du groupe (**AuthGroupFile**) selon la philosophie du serveur Apache.



11.11 Virtual Server

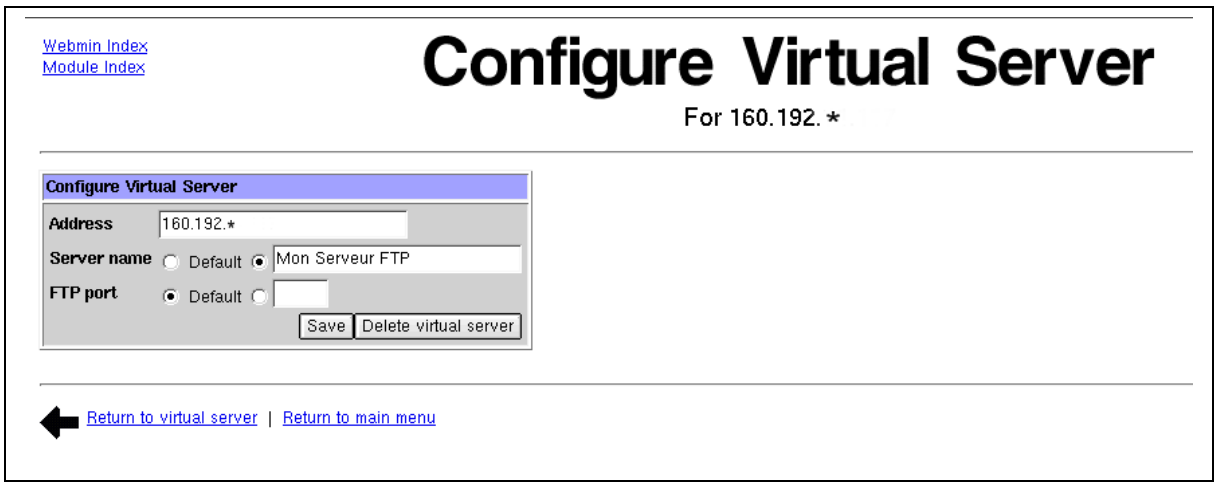
11.11.1. Menu Général

Ce menu général présente beaucoup de similitudes avec les précédents. Nous nous attarderons sur les menus nouveaux à ce niveau de configuration tels que : **Configure Virtual Server**, **Edit Directives** et **Commands Login**.



Le menu *Commands Login* apparaît uniquement dans le menu du deuxième serveur virtuel accessible depuis le menu principal.

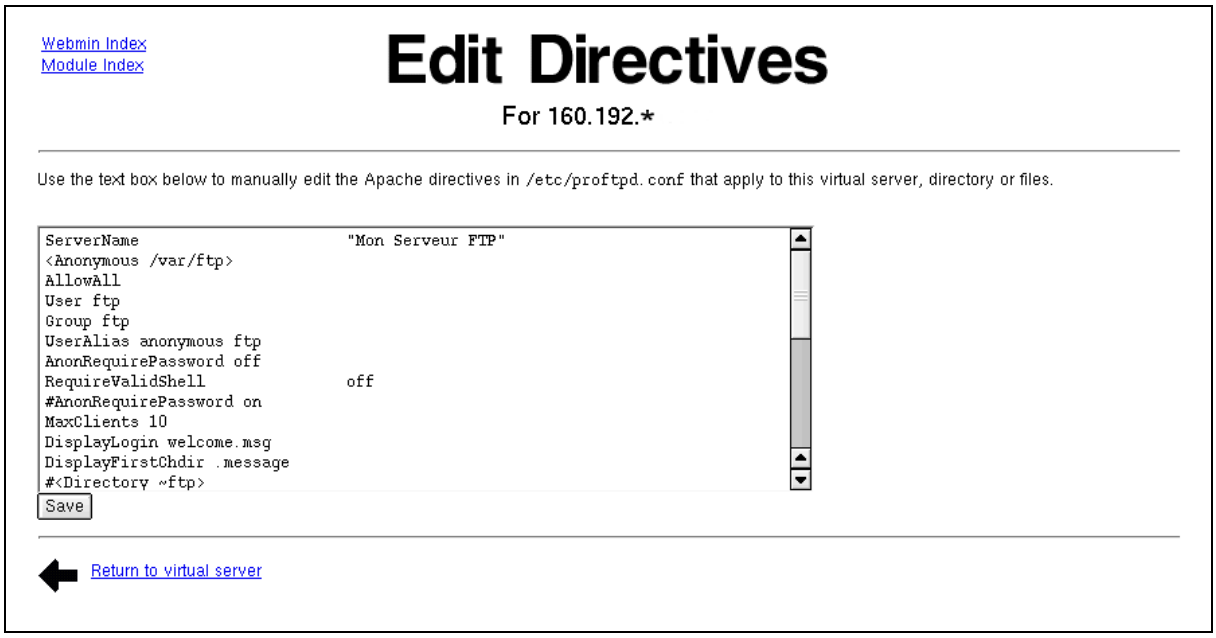
11.11.2. Configure Virtual Server



Ce menu décrit la configuration d'un serveur virtuel uniquement au niveau des identificateurs d'accès : **ServerName, Port**.

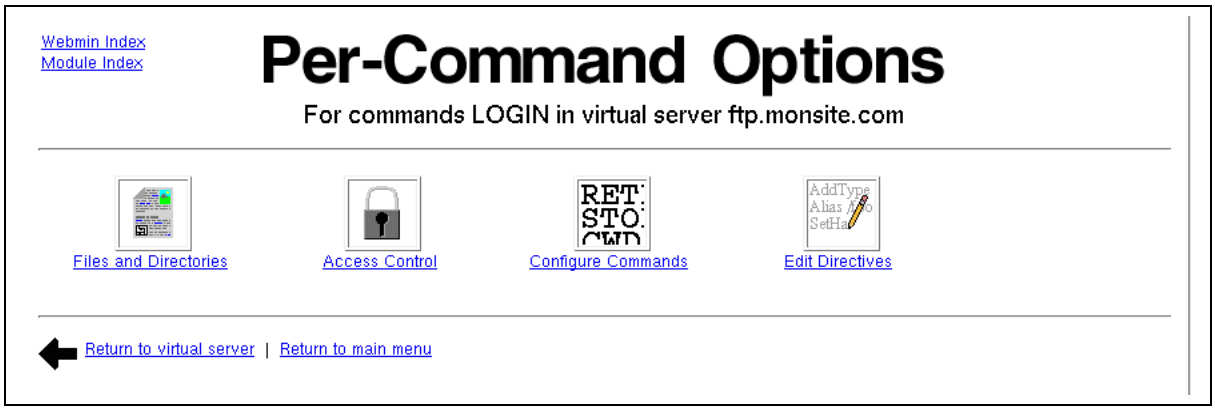
11.11.3. Edit Directives

Ce menu permet d'éditer la partie du fichier de configuration spécifique à la configuration de ce serveur virtuel.

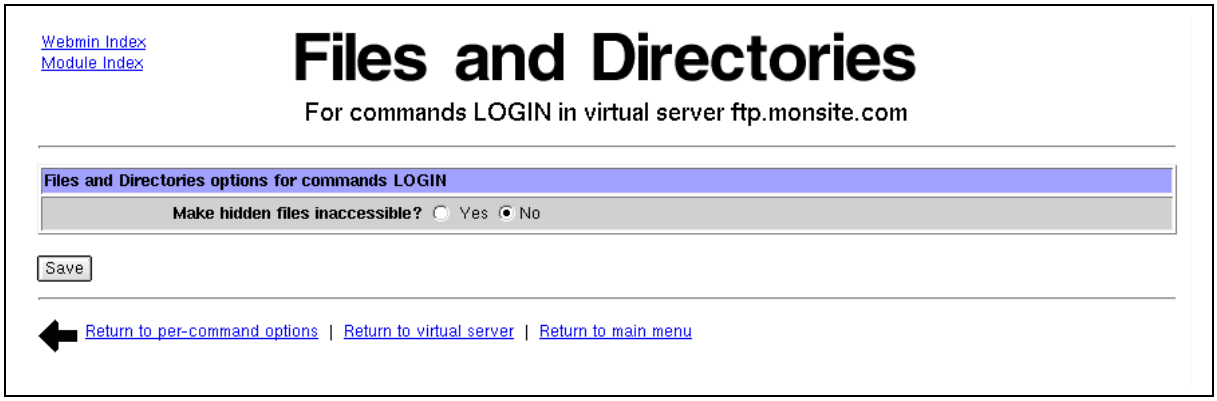


11.11.4. Commands Login

Ce menu donne accès à une nouvelle page de menus secondaires présentée par la copie d'écran suivante.
Seul le menu **Configure Commands** est nouveau.



11.11.5. File and Directories

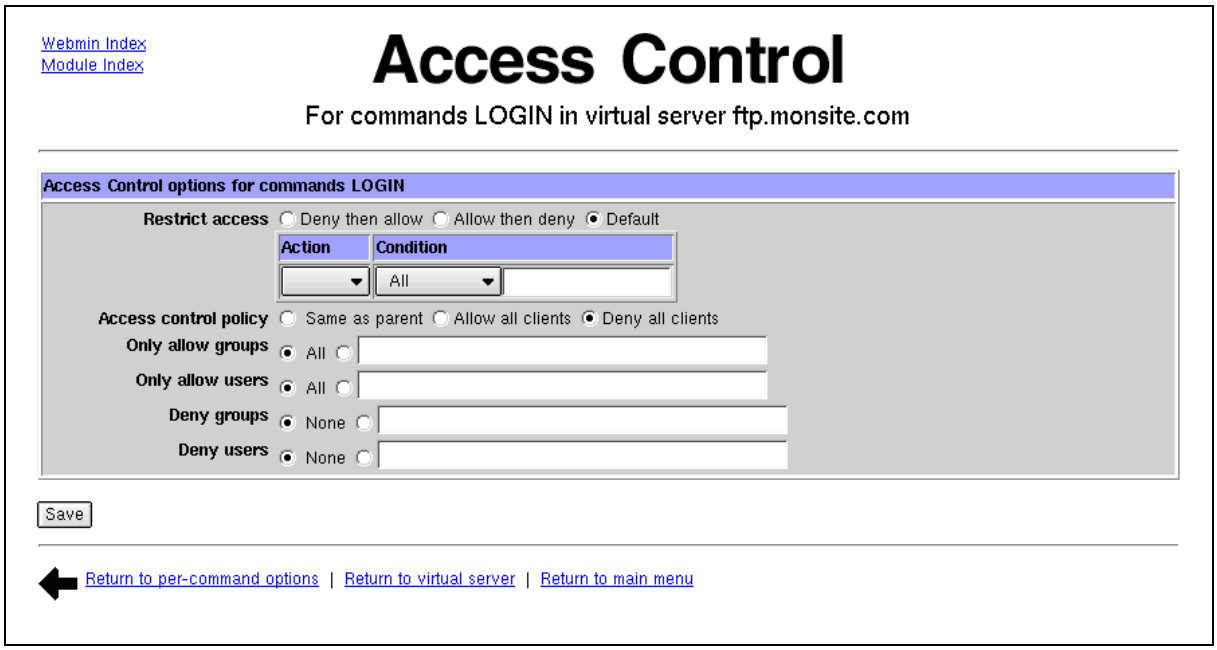


Ce menu présente une seule option : celle de rendre les fichiers cachés accessibles (**IgnoreHidden**) ou non.

11.11.6. Access Control

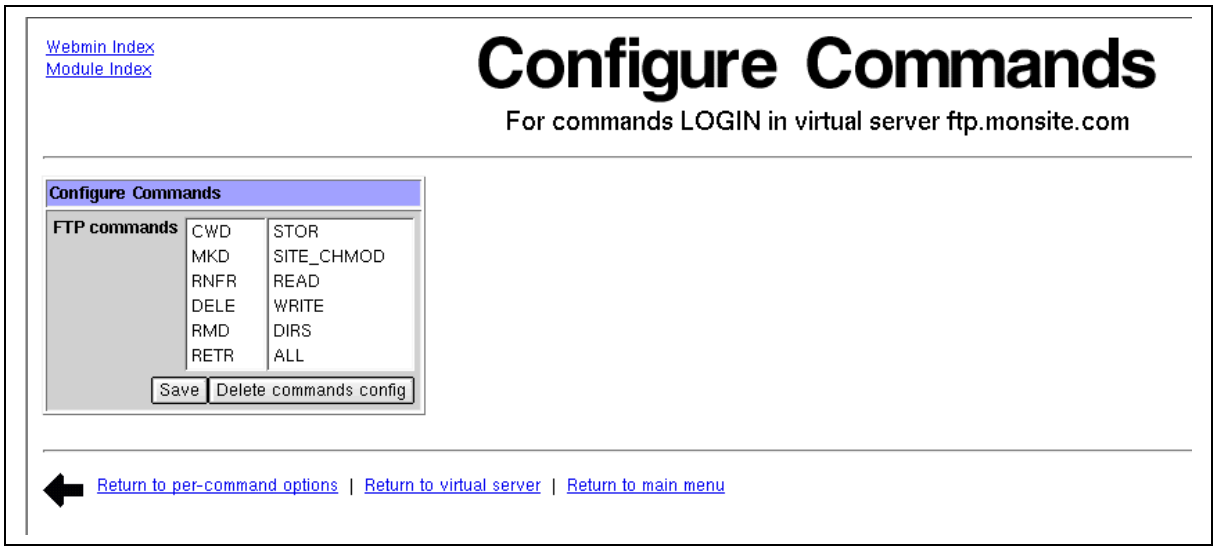
Ce menu présenté ici permet à l'administrateur du site :

- de définir la politique d'accès ou d'interdiction des clients (**Order**), puis définir les exceptions aux règles prédéfinies (**Allow, Deny**) ;
- de faire hériter à ce niveau des règles définies plus haut dans la configuration ;
- de limiter à ce niveau les accès des clients autant au niveau des groupes que des utilisateurs (directives **AllowGroup** et **AllowUser**, **DenyGroup** et **DenyUser**).



11.11.7. Configure Commands

Ce dernier menu décrit les commandes autorisées sur un autre serveur virtuel donné (*ftp.monsite.com*) à l'aide du contexte **LIMIT**.



4. Fichiers de Configuration du Logiciel

Dans le cas où l'administrateur du serveur n'a pas choisi les localisations standards pour les fichiers de configuration du serveur, il peut l'indiquer de manière interne au logiciel Webmin en cliquant sur le lien hypertexte *Module Index* situé en haut à gauche de chaque page (voir copie d'écran précédent).