Installation et configuration de Pure-FTPd

- site officiel : http://www.pureftpd.org/project/pure-ftpd
- tuto 1 (ligne de commande) : http://drpixel.tuxfamily.org/index.php?2006/06/25/8-pure-ftpd-installation-et-configuration-1-2
- avec pureadmin : http://drpixel.tuxfamily.org/index.php?2006/07/02/9-pureadmin-administration-graphique-de-pure-ftpd

Présentation

On peut avoir besoin d'installer un serveur FTP pour de multiples raisons.

Pure-ftpd permet de mettre en place des utilisateurs virtuels, chrootés, avec des options de restrictions propres à chaque utilisateur.

Très simple à configurer, il comporte de nombreuses fonctions, comme :

- limite du nombre d'utilisateurs
- limitation de la bande passante
- chroot des utilisateurs
- support du protocole FXP (File eXchange Protocol)
- support de plusieurs types d'authentification (Unix, PAM, LDAP, MySQL, PostgreSQL, utilisateurs virtuels)
- support des quotas
- limitation de la connexion en fonction du créneau horaire
- limitation du nombre de connexion par IP
- limitation du nombre de connexion par utilisateurs
- log via syslog
- autocréation du dossier personnel
- ratio upload/download
- support de TLS

Téléchargement

http://www.pureftpd.org/project/pure-ftpd/download

Installation

La sécurité d'abord

Le serveur peut fonctionner avec séparation de privilèges pour une sécurité paranoïaque.

Il peut même fonctionner totalement 100% non-root, avec émulation chroot() et comptes virtuels.

La transmission des mots de passe et des commandes en clair peut être évitée : Pure-FTPd supporte optionnellement une connexion avec couche de cryptage SSL/TLS utilisant la bibliothèque OpenSSL.

Pure-ftpd fonctionne sur votre serveur

Le même code source se compile et fonctionne sous Linux, OpenBSD, NetBSD, FreeBSD DragonflyBSD, Solaris, Tru64, Darwin, Irix, HPUX, AIX et iPhone.

Pure-ftpd parle votre langue

Tous les messages du serveur sont traduits en anglais, allemand, roumain, français, polonais, espagnol, danois, néerlandais, italien, portugais brésilien, slovaque, coréen, suédois, norvégien, russe, chinois traditionnel, chinois simplifié, tchèque, turc, hongrois et en catalan.

Un excellent choix pour les débutants

Un débutant peut installer un serveur Pure-ftpd en 5 minutes. Il suffit d'installer le package, de taper “pure-ftpd &” et … c'est tout. Le serveur s'exécute et les clients peuvent commencer à se connecter.

Inutile d'examiner un fichier de configuration long et complexe, où toute erreur pourrait avoir des implications de sécurité et de fiabilité. Pure-ftpd utilise de simples commutateurs de ligne de commande pour activer les fonctionnalités voulues.

On peut limiter le nombre d'utilisateurs simultanés, limiter leur bande passante, cacher des fichiers système (chroot), placer des ratios d'upload/download et modérer les nouveaux uploads. Des messages personnalisés peuvent s'afficher à la connexion en temps (en changeant les fichiers “fortune”) et quand un utilisateur entre dans un nouveau répertoire. Pour éviter de remplir les disques, on peut définir un pourcentage maximal → tout nouvel ajout sera rejeté une fois ce pourcentage est atteint.

Le protocole mis en œuvre est FXP (serveur à serveur). Il peut être disponible pour tout le monde, ou seulement pour les utilisateurs authentifiés.

Pure-ftpd fournit une protection contre les petits malins qui utilisent des outils force-brute qui tentent de découvrir des répertoires cachés. L'accès anonyme est sécurisé par défaut. Par exemple, les utilisateurs ne peuvent pas accéder aux fichiers préfixés d'un point (.Bash_history, .Rhosts, …) sauf activation explicite

Et pour voir qui fait quoi, la commande pure-ftpwho montre dasn une table les sessions actives, la bande passante prise par tous les utilisateurs, quels fichiers ils téléchargent, de là où ils viennent, etc.

Une grande flexibilité pour les fournisseurs de services Internet et d'hébergement

- Les Comptes système peuvent immédiatement avoir un accès FTP. L'authentification via les modules PAM est également pris en charge. Les comptes au-dessous d'un certain uid (par exemple <500 pour les comptes démon) peut être désactivés
- Tous les comptes peuvent être facilement chrootés par défaut. Pour faciliter l'administration, un groupe “de confiance” sans chroot peut être défini.
- Les comptes FTP peut être distincts des comptes système, stockés dans une base de données indépendante. Plusieurs comptes peuvent partager le même identifiant système. Une base de données d'indexation intégrée permet des recherches très rapides. Fonctionne avec plus de 1,5 millions de comptes sur le même serveur. Les comptes système peuvent être copiés sur des comptes FTP virtuels, de sorte que les utilisateurs peuvent avoir différents mots de passe pour l'accès au shell et l'accès FTP.
- L'authentification LDAP est également entièrement prise en charge. Des fonctions de hachage cryptographiques en clair, Crypt, MD5, SMD5, SHA et SSHA sont mises en œuvre. Pure-ftpd a été testé avec succès avec OpenLDAP et iPlanet Directory Server. Il utilise les classes standards des comptes posix.
- Les hachages cryptographiques sécurisés (SMD5, l'ASIS) intégrés peuvent être utilisés avec n'importe quel serveur LDAP, même ceux qui sont peu adaptés pour ces hachages.
- Les infos utilisateurs peuvent également être centralisées dans des bases de données MySQL , avec ou sans les transactions. Toutes les requêtes sont entièrement personnalisables, et les peuvent être construites avec noms d'utilisateurs, adresses de clients distants, adresses IP et ports locaux. Ainsi, des règles complexes d'hébergement peuvent être facilement mises en œuvre, même avec plusieurs serveurs virtuels sur un même hôte et de multiples domaines virtuels avec de nombreux utilisateurs.
- Des méthodes d'authentification multiples peuvent être enchaînées dans un ordre quelconque. Par exemple, les comptes SQL, les annuaires LDAP et les comptes système peut être utilisé en même temps.
- Des méthodes d'authentification personnalisées peuvent être facilement ajoutés. Pure-ftpd supporte les modules d'authentification externes, et la rédaction d'un nouveau backend peut être aussi simple que quelques lignes de script shell
- Pure-ftpd supporte un système de quotas virtuel : les comptes peuvent avoir un quota individuel (nombre maximum de fichiers, taille maximum), même si elles partagent le même uid système.
- On peut limiter la bande passante, avec des réglages distincts pour l'upload et le download.
- Chaque utilisateur peut avoir son quota, son ratio et sa bande passante.
- Chaque utilisateur peut être autorisé de se connecter uniquement à partir d'une gamme spécifique d'adresses IP, ou seulement à son hôte virtuel.
- Chaque utilisateur peut être individuellement limité à son répertoire personnel ou non.
- Chaque utilisateur peut être autorisé à se connecter uniquement pendant une période de temps donnée (par exemple, uniquement pendant les heures de bureau).
- Un système anti-warez empêche les utilisateurs de faire une transaction, s'ils ont trouvé un répertoire public en écriture. Les fichiers appartenant à des utilisateurs ftp anonymes ne peuvent pas être téléchargés (l'administrateur doit les modérer en modifiant leurs permissions, les rendant owner). En outre, les utilisateurs FTP ne peut pas créer des répertoires par défaut pour cacher des fichiers.

- Tout script shell externe peut être appelé après un chargement réussi. Les scanneurs de virus et archiveurs de base de données peuvent facilement être mis en place.
- Un nombre maximum de connexions simultanées à partir de la même adresse IP peut être appliqué pour éviter l'engorgement de la bande passante et les attaques de déni de service.
- Les téléchargements peuvent être refusés si la charge du système est trop élevée.
- Les listings de répertoires peuvent afficher un certain nombre maximum, paramétrable, de fichiers. Les listings récursifs sont entièrement pris en charge, avec une profondeur maximale paramétrable. Ce qui permet de fournir la recherche récursive aux utilisateurs sans fournir de simples déni de service.
- La commande pure-ftpwho fournit en temps réel des rapports de qui fait quoi sur le serveur FTP, y compris l'utilisation de la bande passante. Le résultat peut être une page Web complète, et le programme peut aussi fonctionner comme un programme CGI standard, compatible avec n'importe quel serveur web. Des rapports XML et texte sont également disponibles, ainsi qu'un format compact et facilement analysable pour les scripts shell.
- Les fichiers journaux sont précis et utilisent les syslog standard du système. Des journaux supplémentaires style Apache (CLF) peuvent être produits. Ils sont compatibles avec tous les logiciels de web-statistique. Un format étendu appelé “Stats” est également mis en œuvre, et fonctionne avec des logiciels tiers de statistiques FTP comme FTPStats et ModLogAn. FTPStats fournit des informations statistiques détaillées pour chaque utilisateur.
- Des répertoires Home peuvent être créés à la demande. Ceci est particulièrement utile avec les backends LDAP et SQL : il suffit d'insérer une ligne dans la base de données, et le compte est prêt. Pas besoin de créer un répertoire pour l'utilisateur: il sera automatiquement créé à la première connexion.
- De multiples serveurs virtuels FTP peuvent être hébergés sur le même ordinateur, avec une adresse IP de confiance indépendante pour l'administration
- L'accès aux fichiers préfixés d'un point peut être limité pour que les utilisateurs ne puissent pas lire ou écrire des répertoires .ssh, les fichiers .bash_history, les fichiers .rhosts, etc.
- Des autorisations de sécurité sont appliquées sur les répertoires personnels des utilisateurs. Les clients ne peuvent pas désactiver leurs comptes par erreur avec une commande “chmod 0 /”. La commande “chmod” peut également être totalement désactivée.
- De multiples serveurs pure-ftpd peuvent fonctionner sur le même hôte avec des réglages différents sans aucun conflit.
- Pure-FTPd peut agir en tant que serveur FTP privé et interdire toutes les connexions anonymes quel que soit le compte système “ftp”. Avec un autre commutateur, le serveur peut être seulement anonyme et refuser des connexions aux comptes shell.
- Les liens symboliques peuvent être suivis quand les utilisateurs sont chrootés, même quand cela les mène en dehors de la prison chroot. Cette caractéristique unique permet de configurer facilement des contenus partagés.
- Des alias de répertoires peuvent être activés pour fournir des raccourcis vers des répertoires communs.
- Les uploads sont vraiment atomiques. Le serveur Web ne servira ni images partielles, ni scripts PHP brisés lorsque les fichiers sont téléchargés, même en cas de mise à jour du contenu.

Documentation

Le serveur a été conçu pour être sécurisé dans la configuration par défaut, il n'a pas de vulnérabilité

connue, il est vraiment trivial à mettre en place et il est spécialement conçu pour les noyaux modernes. Il a été porté avec succès sur Linux, FreeBSD, DragonflyBSD, NetBSD, OpenBSD, ISOS, MirBSD, BSDi, Solaris, Darwin, Tru64, Irix, AIX, HPUX et iPhone.

Les caractéristiques comprennent : répertoires chroot()és et / ou répertoires home virtuels chroot()és, domaines virtuels, 'ls' de base, système anti-warez, ports configurables pour les téléchargements passifs, protocole FXP, limitation de bande passante, ratios, authentication basée sur LDAP / MySQL / PostgreSQL, fichiers fortune, fichiers journaux Apache-like, mode autonome rapide, rapport d'état texte / HTML / XML en temps réel, utilisateurs virtuels, quotas virtuels, séparation de privilèges, SSL/TLS et plus.

Compilation

Dans sa forme actuelle, pure-ftpd utilise certains appels système spécifiques à l'OS. Et bien que certains travaux de portabilité vers d'autres systèmes d'exploitation aient été faits, seuls Linux, FreeBSD, NetBSD, OpenBSD, ISOS, MirBSD, BSDi, DragonflyBSD, Darwin, Solaris, Tru64, Irix, HPUX et AIX sont connus pour fonctionner. D'autres systèmes d'exploitation peuvent avoir besoin quelques ajustements. Avec Linux, une distribution moderne devrait être ok.

Étape 1 (optionnel mais recommandé)

Créer un utilisateur particulier, sans privilége et un groupe appelé _pure-ftpd, sans shell valide. Ne pas l'utiliser pour quoi que ce soit d'autre, y compris les utilisateurs virtuels FTP.

```
sudo groupadd _pure-ftpd
sudo useradd -g _pure-ftpd -d /var/empty -s /etc _pure-ftpd
```

Si avoir un utilisateur dont le nom commence par un trait de soulignement vous ennuie, vous pouvez également l'appeler pure-ftpd, sans le trait de soulignement.

Étape 2

Si CDialog ou Xdialog est installé sur le système, essayer la commande suivante pour compiler et installer pure-ftpd :

```
make -f Makefile.gui
```

Sinon ou si vous préférez la manière conventionnelle, c'est ici :

```
./configure
make install-strip
```

Et voila! Le logiciel est maintenant installé dans /usr/local/sbin/pure-ftpd

Étape 3

Pour lancer le serveur, il suffit de taper la commande suivante :

```
/usr/local/sbin/pure-ftpd &
```

Si vous avez installé un paquet binaire (RPM, SLP, Debian),on peut utiliser la commande suivante :

```
/usr/sbin/pure-ftpd &
```

Le serveur est prêt. Il suffit de taper

```
ftp localhost
```

pour le tester.

Pour exécuter automatiquement le serveur au démarrage du système, ajouter la commande précédente dans /etc/rc.d/rc.local ou /etc/rc.d/boot.local. Ne pas oublier le signe '&'.

<note tip>Pour compiler sous Irix, vous devez taper ceci avant de taper ./configure:

```
export CC=cc
export CFLAGS=-I/usr/freeware/include
export LDFLAGS=-L/usr/freeware/lib32
```

Pour compiler sous Solaris, utiliser Make de GNU , pas celui de Solaris. Ensuite, faire :

```
export PATH=/usr/ccs/bin:$PATH
export MAKE=gmake
```

Pour désinstaller Pure-FTPd, faire :

```
./configure
make uninstall
```

</note>

Compilation avancée

Le script “./configure” accepte quelques arguments à ajouter avant la compilation :

commutateurs --without-

- –without-privsep : désactiver la séparation de privilèges (voir notes à ce sujet plus loin), pas recommandé.
- –without-ascii : ne prend pas en charge les transferts 7-bits (ASCII). Si vous avez des clients qui

utilisent les clients Windows pour envoyer des scripts et des fichiers HTML, n'utilisez pas cette option ou ils vont hurler.

- –without-capabilities : s'il existe la bibliothèque de capacités (libcap), Pure-FTPd va essayer de l'utiliser afin d'améliorer la sécurité. Cette option remplace le test d'ignorer la bibliothèque. Essayez ceci si les capacités ne fonctionnent pas correctement sur votre système. libcap peut être téléchargé à partir de ftp://ftp.kernel.org/pub/linux/libs/security/linux-privs/.
- –without-globbing : ne pas inclure le code d'englobement. Cela réduit l'encombrement mémoire, mais les expressions régulières ne fonctionnent plus (des choses comme 'ls *.rpm'). La plupart des gens ne devraient pas utiliser -without-globbing. Globbing est une fonctionnalité intéressante.
- –without-humor : si vous trouvez ce que cette option fait sans regarder le code source, vous êtes un homme chanceux !
- –without-inetd : si vous voulez toujours exécuter Pure-ftpd en mode autonome, activer ce drapeau peut sauver quelques octets de code. Ne pas activer –without-inetd et –without-standalone, car il est impossible de faire tourner un serveur sans l'un d'entre eux. Ces options ne sont pas activées sur les distributions binaires de Pure-FTPd, de sorte que les deux modes inetd-like et autonome sont pris en charge.
- –without-iplogging : ne pas connecter n'importe quelle adresse IP pour protéger la confidentialité, en particulier pour les serveurs politiques.
- –without-nonalnum : test paranoiaque du nom de fichier : n'autoriser que des caractères alphanumériques de base. Ne jamais activer ce commutateur à l'aveuglette, ou vos clients se plaindront.
- –without-unicode : interdire les caractères non-latins. Recommandé si vous n'avez pas de caractères spéciaux dans les noms de fichiers.
- –without-sendfile : sur les noyaux Linux, Solaris, HP-UX et FreeBSD, Pure-ftpd tente de réduire l'utilisation du processeur / mémoire en utilisant un appel système spécial (sendfile). Cela fonctionne très bien avec la plupart des systèmes de fichiers. Toutefois, cette optimisation n'est pas appliquée à tous les systèmes de fichiers dans les noyaux actuels. Des utilisateurs ont signalé que le téléchargement de fichiers avec Pure-FTPd a échoué avec SMBFS (Samba) sur FreeBSD et tmpfs et NTFS sous Linux (l'erreur signalée par le serveur est “broken pipe” ou “Error during write to data connection”). Si vous avez l'intention de fournir des fichiers sur ces systèmes de fichiers, vous devez utiliser le commutateur –without-sendfile pour permettre un contournement. Il a également été rapporté que systèmes PA-RISC Linux ont besoin de ce drapeau.
- –without-shadow : ignorer les mots de passe masqués, même s'ils sont détectés automatiquement. Généralement une mauvaise idée, sauf si vous utilisez PAM, LDAP ou SQL. Pure-ftpd gère les dates d'expiration (à la fois pour les comptes et mots de passe).
- –without-standalone : le serveur FTP peut normalement fonctionner en mode autonome (sans super-serveur). Si vous n'avez pas besoin de cette fonctionnalité et si vous voulez économiser quelques octets de code, ajouter cette option. Un super-serveur comme g2s, xinetd ou tcpserver sera nécessaire pour exécuter le service. Mais le mode autonome est le mode recommandé.
- –without-usernames : ne jamais afficher les noms d'utilisateur et de groupe dans les listes de répertoires, mais seulement les UID et GID. Cela améliore la sécurité et les performances, mais certaines personnes trouvent cela peu convivial.

Autres notes

D'autres options autoconf traditionnelles sont bien sûr reconnu, en particulier :

- –prefix= pour changer le préfixe d'installation (par défaut “/usr/local/”)
- –sysconfdir= pour changer le répertoire des fichiers de configuration (par défaut, “/etc” sauf si vous avez spécifié un préfixe avec –prefix)
- –localstatedir= pour changer le répertoire des fichiers exécutables (par défaut “/var” même si vous avez spécifié un préfixe avec –prefix)

Pour info, les paquetages RPM binaires de Pure-FTPd sont configurés avec la ligne de commande suivante :

```
./configure --with-everything --with-paranoidmsg --without-capabilities --
with-virtualchroot
```

Les paquetages RPM sont également compilés avec –without-pam pour améliorer leur portabilité.

Installation autonome

Il s'agit de la méthode recommandée pour démarrer le serveur.

Sauf si vous compilez le serveur avec “–without-standalone”, lancer le serveur est aussi facile que de taper :

```
/usr/local/sbin/pure-ftpd &
```

Dans les exemples suivants, nous supposons que le fichier 'pure-ftpd' est situé dans /usr/local/sbin. Il s'agit de la valeur par défaut si vous avez compilé le serveur depuis l'archive du code source. Mais comme je l'ai dit plus tôt dans ce document, si vous avez installé un paquet binaire (RPM, SLP, deb, tgz), le serveur peut être installé dans /usr/sbin/. Il suffit donc de remplacer '/usr/local/sbin/pure-ftpd' par '/usr/sbin/pure-ftpd'.

Lorsque la commande précédente est exécutée, le serveur écoute les connexions entrantes sur chaque interface, toutes les adresses IP et le port FTP standard (21). Si votre système a des adresses IPv6, ils devraient travailler ainsi.

Maintenant, si vous voulez écouter une connexion entrante sur un port non standard, suffit d'ajouter “-S” et le numéro de port :

```
/usr/local/sbin/pure-ftpd -S 42
```

Les noms de services sont également autorisés ('S-smtp “et le démon acceptera les connexions sur le port SMTP (25). Très rare, mais il faut que tout le monde soit content, même les esprits perturbés).

Et maintenant, si votre système possède de nombreuses adresses IP et que vous voulez que le serveur FTP soit joignable sur une seule de ces adresses, disons 192.168.0.42 ? Il suffit d'utiliser la ligne de commande suivante :

```
/usr/local/sbin/pure-ftpd -S 192.168.0.42,
```

La virgule finale est importante, ne l'oubliez pas. En fait, c'est un raccourci pour :

```
/usr/local/sbin/pure-ftpd -S 192.168.0.42,21
```

Si vous préférez des noms d'hôte aux adresses IP, c'est votre choix :

```
/usr/local/sbin/pure-ftpd -S ftp.example.com,21
```

Les adresses IPv6 sont bien sûr prises en charge.

Avec des lignes de commande précédentes, le serveur va tourner dans la configuration par défaut. Les connexions FTP anonymes seront autorisées s'il ya un compte système appelé 'ftp' et chaque utilisateur de votre système sera en mesure d'accéder au serveur FTP en utilisant son couple login / password.

Si vous devez modifier cette configuration par défaut, d'autres lignes de commande options peuvent être ajoutées. Par exemple:

```
/usr/local/sbin/pure-ftpd -c 50 &
```

ou

```
/usr/local/sbin/pure-ftpd -S ftp.example.com,21 -c 50 &
```

Et seulement 50 connexions simultanées seront autorisées. Pour découvrir quelles options sont disponibles passer directement au chapitre «Options» ci-dessous. Si le serveur fonctionne parfaitement pour vous en mode autonome, vous n'avez pas besoin de lire le chapitre suivant sur les super-servers. Mais lisez les options. '-m' and '-C' sont recommandées. '-D' est également un bon choix si vous (ou vos clients) utiliser des clients broken. S'il vous plaît lisez la suite.

Lorsque vous exécutez 'ps auxw|grep pure-ftpd', le résultat ressemble à ceci :

```
root     15211  0.1  0.3  1276  452 ?        S    13:53   0:00 pure-ftpd
[SERVER]
root     15212  0.1  0.5  1340  672 ?        S    13:54   0:00 pure-ftpd
[IDLE]
root     15214  0.0  0.5  1340  672 ?        S    13:56   0:00 pure-ftpd
[DOWNLOADING]
```

- [SERVER] est le serveur principal. Si vous tuez ce processus, le serveur va quitter après la connexion suivante.
- [IDLE] montre un client sans activité de transfert.
- [DOWNLOADING] montre un client en train de télécharger un fichier.
- [UPLOADING] montrent une client en train d'uploader un fichier.

Pour faciliter les scripts, le fichier '/var/run/pure-ftpd.pid' est créé et il contient toujours le PID du processus serveur principal.

Si vous voulez arrêter le serveur, il vous suffit de tuer les processus :

```
pkill pure-ftpd
```

Bien sûr, ne pas utiliser -9 sauf si le serveur est complètement coincé. -9 Ne laisse pas la moindre chance au processus pour sortir proprement et ne devrait jamais être utilisée, sauf si il n'y a absolument rien d'autre à faire.

< 50% 10em ->	
code	**action**
-0	quand un fichier est uploadé et qu'il y a déjà une version précédente du fichier avec le même nom, l'ancien fichier ne sera ni supprimé, ni tronqué. L'upload aura lieu dans un fichier temporaire et une fois le téléchargement terminé, le passage à la nouvelle version sera atomique. Par exemple, quand un grand script PHP est cours d'upload, le serveur Web fournit toujours l'ancienne version et passera immédiatement à la nouvelle dès que le dossier complet aura été transféré
-1	enregistre le PID de chaque session dans la sortie syslog
-4	n'écouter que les connexions IPv4
-6	ne pas écouter pas IPv4, écouter seulement IPv6
-a <gid>	Les utilisateurs authentifiés pourront accéder à leur répertoire personnel et rien d'autre (chroot). Ceci est particulièrement utile pour les utilisateurs sans accès au shell, par exemple, les services WWW-hosting partagés par plusieurs clients. Seul membre du groupe numéro <gid> auront libre accès au système de fichiers entier. Donc ajoutez un groupe «staff», «admin» ou «ftpadmin” et mettez-y vos utilisateurs de confiance. <gid> est un numéro de groupe NUMERIQUE, et non pas un nom de groupe. Cette fonctionnalité est principalement conçu pour les utilisateurs du système, et non pour les utilisateurs virtuels