SERVEUR FTP SOUS LINUX

Si vous souhaitez stocker et récupérer régulièrement des fichiers plus volumineux sur un serveur, vous avez alors besoin d'une technologie de transmission appropriée. Parmi les solutions possibles, le **transfert de fichiers via File-Transfer-Protocol (FTP)** reste l'une des plus populaires. Outre le transport des données via les **ports TCP 20 et 21**, le protocole de transmission pour les réseaux IP agissant sur la couche applicative offre aussi à l'utilisateur la possibilité de créer des répertoires et de les structurer selon les besoins. Comme vous n'avez besoin d'accéder au serveur FTP qu'avec un logiciel client et que la mise en réseau des deux composants **via Internet** est suffisante à cette fin, la technologie de transfert est souvent utilisée pour charger et télécharger des fichiers vers des espaces Web.

En tant que locataire de telles ressources du site Web, vous recevez généralement des données de connexion spécifiques qui peuvent être utilisées pour **se connecter au serveur FTP du fournisseur d'hébergement**. En tant qu'application auto-hébergée, vous pouvez bien entendu utiliser la **technologie de transfert de fichiers** en installant et en configurant votre propre serveur en conséquence. Ce guide explique comment configurer un tel serveur FTP sous linux (y compris le chiffrement TLS).

Installer un serveur FTP

Avant de pouvoir vous familiariser avec l'installation et la configuration, vous devez tout d'abord trouver et installer le logiciel serveur approprié. Pour Linux, il existe différents serveurs FTP, dont la plupart sont **open source** et se trouvent généralement dans la gestion des paquets de la distribution respective. Une des applications les plus populaires est ProFTPD, sous licence GPL et qui est très extensible grâce à sa conception modulaire. Le fichier de configuration principal fonctionne sur la base de directives et de groupes de directives qui devraient sembler familiers à tout administrateur habitué aux serveurs Web Apache. Debian/ubuntu maintient aussi **par défaut ProFTPD dans le répertoire logiciel**. L'installation s'effectue donc de manière habituelle via le terminal et avec la commande suivante :

```
sudo apt-get install proftpd
```

Pour terminer l’installation, il est nécessaire de décider ensuite si vous voulez utiliser **ProFTDP en mode serveur (standalone)** ou en tant que **service contrôlé par inetd**. Dans le premier cas, le serveur FTP se charge lui-même des requêtes entrantes.

Dans la deuxième variante « Superserveur » **inetd/xinetd** reçoit les requêtes et les transmet au serveur FTP (ce qui n’est intéressant que si très peu de trafic FTP est alors attendu).

Les étapes importantes de la configuration

Après l'installation, vous pouvez commencer à configurer ProFTPD. Le fichier de configuration **proftpd.conf** qui est nécessaire, se trouve dans /etc/proftpd/. Pour l'éditer, il suffit de l'ouvrir avec **l'éditeur de votre choix.** Avec le programme standard Debian/ubuntu nano, cela peut être fait par exemple en utilisant la commande de terminal suivante :

```
sudo nano /etc/proftpd/proftpd.conf
```

Dans les différentes lignes, vous trouverez les fonctions et paramètres les plus importants du serveur FTP sous Debian. **Chaque composant reçoit une ligne séparée et nécessite des valeurs fixes** : par exemple, si une fonction doit être utilisée, la valeur « *on* » (fonction activée) ou la valeur « *off* » (fonction désactivée) sont possibles. De plus, le **signe(#)** peut être placé devant une ligne pour « commenter ». Le serveur ProFTDP ignore alors complètement la ligne, donc cette notation est une autre option pour **désactiver les fonctions.** Le but principal du mot dièse ou hashtag, cependant, est de faire des commentaires sur les différents paramètres afin d’améliorer la lisibilité du fichier **proftpd.conf.**

Configurations de base : nom de serveur, répertoire FTP etc.

Avant d'entrer dans le détail de la configuration de votre serveur FTP, vous devez tout d'abord ajuster la configuration de base. Il s'agit de paramètres élémentaires tels que la spécification du **nom d'hôte ou du répertoire du serveur** que vous souhaitez mettre à disposition pour le **chargement et le téléchargement des fichiers**. Vous disposez aussi de diverses options de configuration qui se réfèrent à des utilisateurs FTP potentiels, comme le montre l'exemple de configuration suivant :

 # Spécification du nom d'hôte et du message de bienvenue

```
ServerName  "nom d'hôte/adresse IP"
DisplayLogin  "La connexion au serveur FTP sous Debian s'est effectuée
avec succès !"

# Instructions générales de connexion
<Global>
  # Autoriser l'accès uniquement avec les interfaces systèmes, qui
sont dans /etc/shells définies
  RequireValidShell  on
  # Refuser la connexion root
  RootLogin  off
  # Spécifie le répertoire FTP auquel l'utilisateur est autorisé à
accéder
  DefaultRoot  état
</Global>

# Définir les utilisateurs/groupes d'utilisateurs autorisés pour la
connexion FT
<Limit LOGIN>
  # L'enregistrement n'est possible que pour les utilisateurs du
groupe de référence ftpuser
  # Au lieu d'une longue liste, le groupe autorisé est simplement nié
(!)
  DenyGroup  !ftpuser
</Limit>
```

Dans cette configuration de base, les utilisateurs ont accès à un répertoire spécifique. Cela a du sens, par exemple, s'ils sont impliqués dans la

maintenance d'un site Web et ont donc besoin de **droits d'accès étendus**. Cependant, si la fonction du serveur FTP Linux est d'offrir aux utilisateurs un seul emplacement pour leurs fichiers, vous devez configurer ProFTPD de sorte que l'accès au **répertoire d'origine** soit restreint :

 # Autoriser uniquement les utilisateurs à accéder à leur répertoire d'origine

```
DefaultRoot  ~
```

Créer un utilisateur FTP

Lorsque vous créez un nouvel utilisateur ProFTPD, vous devez toujours définir */bin/false* comme l'interface système (Shell) de connexion. De cette façon, vous vous assurez que l'utilisateur ne peut accéder qu'au serveur FTP et non à l'ensemble du système. Utilisez la commande de terminal suivante pour entrer */bin/false* dans le fichier des interfaces système (Shells) autorisées :

sudo echo "/bin/false" >> /etc/shells

Vous pouvez ensuite créer un premier utilisateur :

sudo adduser user1 --shell /bin/false --home /home/user1

Dans cet exemple, vous créer un compte utilisateur avec le nom « user1 » et créez son répertoire d'origine dans la même étape. Enfin, attribuez un **mot de passe** pour le nouveau compte utilisateur et confirmez le profil. Pour que cet utilisateur nouvellement crée puisse se connecter au serveur FTP de Debian et télécharger des fichiers dans son répertoire exclusif, spécifiez enfin son répertoire d'origine dans le fichier **proftpd.conf** :

<Directory /home /user1>

```
  Umask 022
  AllowOverwrite off
  <Limit LOGIN>
    AllowUser user1
    DenyAll
  </Limit>
  <Limit ALL>
```

```
    AllowUser user1
    DenyAll
  </Limit>
</Directory>
```

Cet exemple de code restreint le répertoire de plusieurs façons pour alors en faire un référentiel pour les fichiers d'user1 :

avec la commande **Umask(022),** le propriétaire du répertoire reçoit d'abord tous les droits. Les autres utilisateurs ne peuvent lire les fichiers et ne les exécuter que si le propriétaire accorde l'autorisation nécessaire.

La directive *AllowOverwrite* est désactivée pour éviter que les données déjà sauvegardées ne soient écrasées lors du téléchargement de fichiers. Enfin, **la connexion FTP (Limit LOGIN)** et **l'exécution des commandes FTP** (Limit ALL) sont **bloquées** pour tous les utilisateurs à l'exception d'user1.

Autoriser l'accès anonyme

Si vous désirez configurer votre serveur FTP sous linux pour qu'il serve de serveur de téléchargement public, dans la majorité des cas, vous souhaitez alors aussi que les utilisateurs **puissent accéder aux fichiers hébergés de manière anonyme**. Pour cela, utilisez tout d'abord *chmod* pour définir les droits d'accès nécessaires pour le **répertoire de téléchargement** suivant, que nous avons nommé ***/home/ftpdownload*** pour l'exemple :

```
sudo chmod 755 -R /home/ftpdownload
```

Le **propriétaire du répertoire** a donc **tous les droits (7 = lire, écrire et exécuter)**, alors que les utilisateurs du groupe et tous les autres utilisateurs ne peuvent que **lire et exécuter (5)**. Une fois les permissions définies, l'accès anonyme peut alors être configuré dans le fichier proftpd.conf :

```
<Anonymous ~ftp>
User  ftp
```

```
Group  ftpgroup

# Profils de connexion possibles pour Clients
UserAlias  anonymous  ftp

# Masquer les propriétés des utilisateurs et des groupes et
maximiser le nombre de Client
DirFakeUser on ftp
DirFakeGroup on ftp
RequireValidShell  off
MaxClients  10

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

Pour **se connecter au serveur FTP sous linux** avec le profil *ftp*, vous devez l'ajouter au groupe *ftpuser* :

sudo adduser ftp ftpgroup

Configuration du chiffrement SSL/TLS

Le protocole FTP transfère à la fois les informations de connexion et les données envoyées e**n texte clair**. Si vous souhaitez configurer un serveur ProFTPD privé qui ne doit pas être accessible à tous, il est donc recommandé de **chiffrer la connexion**. La solution la plus répandue est le chiffrement via SSL/TLS, que vous pouvez mettre en place en peu d'efforts et en utilisant le **logiciel gratuit OpenSSL**. La boîte à outils cryptographiques est incluse et déjà installée par défaut dans la gestion des paquets Debian. De manière alternative, l'installation peut être réalisée de la manière habituelle :

```
apt-get install openssl
```

Étape 1 : générer le certificat et la clef

Utilisez ensuite OpenSSL, pour créer un certificat. Comme vous devez le stocker quelque part, créer d'abord **le dossier approprié dans le répertoire ProFTPD** :

```
mkdir /etc/proftpd/ssl
```

Générez maintenant le **certificat** (*proftpd.cert.pem*) et la **clef** (*proftpd.key.pem*) avec une durée de vie d'un an pour votre serveur FTP Linux en spécifiant cet emplacement et en utilisant la commande suivante :

```
openssl req -new -x509 -days 365 -nodes -out /etc/proftpd/ssl/proftpd.cert.pem -keyout /etc/proftpd/ssl/proftpd.key.pem
```

De plus, certains renseignements sont nécessaires pour enregistrer correctement le certificat :

- **Country Name(2 letter code)** : code de pays, par exemple „FR“ pour la France
- **State or Province Name(full name)** : Région/Département, par exemple „Alsace“
- **Locality Name (eg, city)** : ville, par exemple „Strasbourg“

•**Organization Name (eg, company)** : nom de l'entreprise ou votre nom

•**Organizational Unit Name (eg, company)** : indication du département (si disponible), par exemple „IT“

•**Common Name (eg, YOUR name)** : indication du domaine à protéger, par exemple „ftp.example.com.“

•**Email Address** : adresse email

Étape 2 : Activer SSL/TLS dans ProFTPD

Après avoir créé votre propre certificat et votre clef privée, vous devez activer la technologie de chiffrement pour le serveur ProFTPD. Pour cela, le logiciel du serveur FTP sur Debian fournit le module **mod_tls**, qui est installé par défaut mais désactivé. Pour l'activation, des ajustements supplémentaires dans le fichier proftpd.conf sont nécessaires. Ouvrez le fichier de configuration et recherchez l'entrée suivante :

```
<IfModule mod_tls.c>
  TLSEngine  off
</IfModule>
```

Assigner la valeur „on“ à la directive TLS Engine et étendre enfin la section comme suit :

```
<IfModule mod_tls.c>
  TLSEngine      on
  TLSLog        /var/log/proftpd/tls.log
  TLSProtocol      TLSv1 TLSv1.1 TLSv1.2
  TLSRSACertificateFile    /etc/proftpd/ssl/proftpd.cert.pem
  TLSRSACertificateKeyFile  /etc/proftpd/ssl/proftpd.key.pem
  TLSVerifyClient      off
  TLSRequired      on
</IfModule>
```

De cette manière, vous avez non seulement activé le chiffrement SSL/TLS pour votre serveur FTP sur Debian, mais vous avez aussi effectué les paramètres les plus importants au cours de la même étape. **Le fichier log pour l'enregistrement des connexions FTP** (*TLSLog*) est défini de la même manière que les chemins d'accès au certificat (*TLSRSACertificateFile*) et à la clef (*TLSRSACertificateKeyFile*). Les versions de protocole possibles (protocole TLS) sont aussi indiquées. Les deux dernières lignes signifient enfin que le module ne vérifie pas les certificats présentés par le client

(*TLSVerifyClient*) et que le **chiffrement est une condition de base** pour établir une connexion (*TLSRequired*). Après le redémarrage du serveur ProFTPD, les nouveaux paramètres prennent effet :

```
sudo /etc/init.d/proftpd restart
```

Étape 3 : se connecter au serveur ProFTPD via SSL/TLS

Si vous avez activé SSL/TLS pour ProFTPD (comme recommandé dans ce tutoriel sur le serveur FTP sous Debian), les utilisateurs ont besoin **d'un client FTP qui supporte les connexions chiffrées**. L'un des représentants les plus importants est **FileZilla**, qui n'est pas seulement disponible pour Debian et d'autres distributions Linux, mais aussi pour macOS et Windows. Ainsi, le programme open source est la solution optimale pour vous et tous les autres utilisateurs d'**accéder au serveur FTP à partir de différentes plateformes**.

Dans le gestionnaire de serveur de FileZilla, vous spécifiez la variante **FTPS** sauvegardée („*FTP par TLS/SSL explicite*“) au lieu de FTP lors de la sélection du type de serveur.

Il est également nécessaire d'accepter le certificat la première fois que vous vous connectez alors au serveur.

Pour résoudre ce problème, vous pouvez utiliser différentes options d'analyse :

1. Vérification de l'exécution du serveur ProFTPD :

```
sudo service proftpd status
```

2. Tester si le serveur ProFTPD écoute sur le port 21 pour enregistrer les requêtes FTP entrantes :

```
sudo netstat -tlp|grep proftpd
```

3. Vérifier les messages d'erreur dans le log ProFTPD :

```
sudo tail -20 /var/log/proftpd/proftpd.log
```

4. Vérifier les messages d'erreur dans le log TLS :

```
sudo tail -20 /var/log/proftpd/tls.log
```

5. Tests de connexion sur le port 21 avec telnet :

```
sudo telnet 192.0.2.10 21
```

6. Tests de connexion sur le port 21 avec TLS:

```
sudo openssl s_client -connect 192.0.2.10:21 -stattls ftp
```