

Intro proftpd

Qu'est ce que proftpd?

ProFTPd est un logiciel libre qui vous permet d'installer un serveur FTP sur votre système d'exploitation. Un serveur FTP (File Transfer Protocol) permet à des utilisateurs d'accéder à vos fichiers pour les télécharger ou les envoyer, à distance et via un réseau.

Installation proftpd

Après avoir créer un container ftp avec l'adresse Ip suivante : 10.31.200.20, nous allons installer Proftpd.

Pour installer proftpd il faut lancer la commande :

```
apt get install proftpd
```

Création du compte std

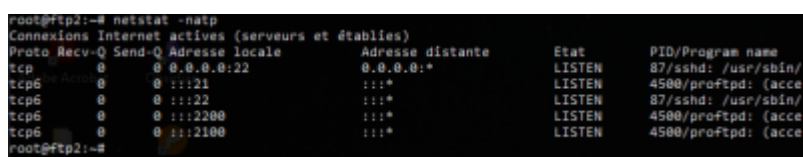
Il faut maintenant créer un compte std afin qu'on puisse se connecter au serveur depuis filezilla.

```
adduser std
```

Vérification du protocole de transport

Pour vérifier le protocole de transport par défaut du service proftpd il faut taper la commande suivante:

```
netstat -natp
```



```
root@ftp2:~# netstat -natp
Connexions Internet actives (serveurs et établies)
Proto Recv-Q Send-Q Adresse locale Adresse distante Etat PID/Program name
tcp        0      0 0.0.0.0:22          0.0.0.0:*        LISTEN 87/sshd: /usr/sbin/
tcp6       0      0 :::21             :::*             LISTEN 4500/proftpd: (acce
tcp6       0      0 :::22             :::*             LISTEN 87/sshd: /usr/sbin/
tcp6       0      0 :::2200           :::*             LISTEN 4500/proftpd: (acce
tcp6       0      0 :::2100           :::*             LISTEN 4500/proftpd: (acce
root@ftp2:~#
```

Nous constatons que le port par défaut de ProFtpd est le port 21 qui est un port bien connu utilisé pour établir une connexion de contrôle entre un client FTP et un serveur FTP.

Configuration du fichier proftpd.conf

Présentation du fichier Proftpd.conf

Le fichier **proftpd.conf** est le fichier de configuration principal du serveur FTP ProFTPd. Il est généralement situé dans le répertoire `/etc/proftpd/`. Ce fichier contient toutes les directives et options qui **définissent le comportement du serveur FTP**, telles que :

- **Les ports d'écoute:** Vous pouvez spécifier les ports sur lesquels le serveur FTP doit écouter les connexions entrantes.

- **Les utilisateurs et les groupes:** Vous pouvez définir les utilisateurs et les groupes qui ont accès au serveur FTP, ainsi que leurs droits et permissions.
- **Les répertoires accessibles:** Vous pouvez définir les répertoires auxquels les utilisateurs peuvent accéder via le serveur FTP.
- **Les options de sécurité:** Vous pouvez configurer divers paramètres de sécurité, tels que le type d'authentification et le cryptage des données.
- **Les modules:** Vous pouvez activer ou désactiver des modules qui ajoutent des fonctionnalités supplémentaires au serveur FTP.

Ce qui doit être configuré

Notre fichier de configuration `proftpd.conf` situé dans le chemin `/etc/proftpd/proftpd.conf` est configuré de sorte que :

- Le fichier de configuration de serveur virtuel `virtuals.conf` soit activé
- Les utilisateurs soient enfermés dans leurs répertoires par défaut avec la ligne `DefaultRoot`
- Les nouveaux fichiers et répertoires créés par les utilisateurs sur le serveur FTP soient définis par défauts avec la ligne `Umask`
- Le droit aux utilisateurs d'écraser des fichiers existants lors du transfert de fichiers vers le serveur FTP avec la ligne `AllowOverwrite`
- Le journal des transferts de fichiers `xferlog.log`
- Le journal des événements du serveur `proftpd.log`
- Qu'un autre fichier de configuration qui contient des directives pour les modules ProFTPD soit inclus, le fichier `modules.conf`

Configuration du fichier `proftpd.conf`

Rendons-nous dans le fichier `proftpd.conf` en tapant cette commande :

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

Nous sommes maintenant dans le fichier de configuration principal de `proftpd`, nous allons le configurer comme ceci:

Informations générales

- **ServerAdmin `iyansagnediagne@beaupeyrat.fr`:** Adresse email de l'administrateur du serveur FTP.
- **ServerName `"STD server"`:** Nom du serveur FTP affiché aux clients.
- **ServerType `standalone`:** Indique que le serveur fonctionne en mode autonome (démon).

Options de connexion

- **DeferWelcome `off`:** Messages de bienvenue affichés immédiatement à la connexion.
- **TimeoutNoTransfer `600`:** Déconnexion après **600 secondes d'inactivité de transfert**.
- **TimeoutStalled `600`:** Déconnexion après 600 secondes de blocage du transfert.
- **TimeoutIdle `1200`:** Déconnexion après 1200 secondes d'inactivité totale.
- **DisplayLogin `welcome.msg`:** Affiche le fichier `"welcome.msg"` à la connexion.
- **DisplayChdir `.message true`:** Affiche le fichier `".message"` lors du changement de répertoire.
- **ListOptions `"-l"`:** Option d'affichage des listes de répertoires avec format long (`"-l"`).
- **DefaultRoot `~`:** Répertoire personnel défini comme répertoire racine par défaut pour les

utilisateurs.

- **Port 21:** Port d'écoute du serveur FTP (port standard 21).
- **MaxInstances 30:** Nombre maximum d'instances simultanées du processus ProFTPD (30 par défaut).

Configuration de l'utilisateur

- **User std:** Nom d'utilisateur exécutant le processus ProFTPD.
- **Group nogroup:** Groupe associé au processus ProFTPD (nogroup par défaut).
- **Umask 022 022:** Permission par défaut attribuée aux nouveaux fichiers et répertoires (022 pour plus de sécurité).

Transfert et journaux

- **AllowOverwrite on:** Autorise l'écrasement de fichiers existants lors du téléversement.
- **TransferLog /var/log/proftpd/xferlog:** Fichier journalisant les transferts de fichiers.
- **SystemLog /var/log/proftpd/proftpd.log:** Fichier journalisant les événements du serveur.

Inclusion de fichiers de configuration supplémentaires

- **#Include /etc/proftpd/tls.conf:** Inclusion optionnelle de la configuration TLS (décommenter pour activer le chiffrement).
- **Include /etc/proftpd/virtuals.conf:** Inclusion du fichier de configuration des serveurs FTP virtuels.



Pour appliquer ces modifications, il faut redémarrer le service proftpd s'il est lancé en mode démon.

```
systemctl restart proftpd
```

Enfermer les utilisateurs

Pourquoi enfermer les utilisateurs

Enfermer les utilisateurs dans leurs répertoires sur un serveur FTP **renforce la sécurité en limitant leur accès et facilite l'organisation des fichiers**. Imaginez des casiers individuels pour chaque utilisateur, plus de désordre et plus de contrôle ! Ceci est fait dans le fichier de configuration proftpd.conf avec la ligne **DefaultRoot ~**

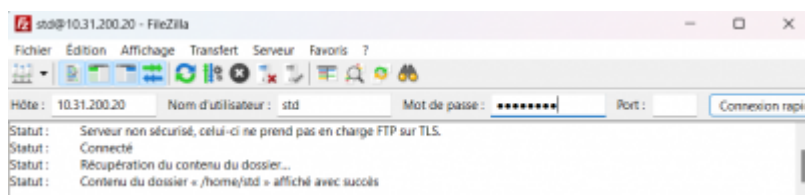
Vérifications

Pour vérifier que les utilisateurs sont bien enfermés dans leurs répertoires personnels, nous le vérifions dans l'application **filezilla**

FileZilla est un outil gratuit qui vous permet de transférer des fichiers d'un ordinateur à un autre via internet. Imaginez-le comme une valise virtuelle que vous utilisez pour envoyer et recevoir des

fichiers sur un serveur distant. Lien pour le télécharger : <https://filezilla-project.org/download.php>

Rendons nous sur filezilla et connectons nous à l'utilisateur **std** avec l'adresse ip du serveur ftp suivante : **10.31.200.20**, après avoir saisi le mot de passe, et le port **21** qui est le port par défaut du serveur ftp nous sommes connecté au serveur ftp sous l'utilisateur std



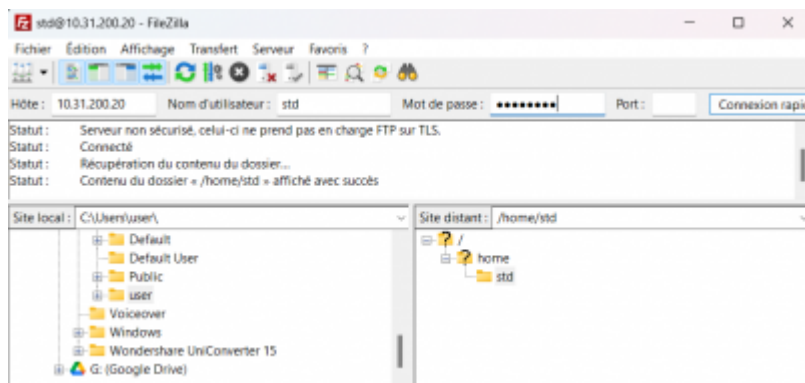
La connexion au serveur ftp2 est bien établie ici, en bas à gauche nous observons que le répertoire par défaut est le répertoire / et qu'on ne peut pas remonter au répertoire parent, c'est grâce à la ligne `DefaultRoot ~` dans le fichier `proftpd.conf`

```
# Use this to jail all users in their homes
DefaultRoot ~
```

Commentons cette ligne et redémarrons proftpd pour observer ce qui va se passer.

```
# Use this to jail all users in their homes
#DefaultRoot ~
```

Le système vient de redémarrer et une reconnexion est effectuée sur filezilla, nous observons que maintenant, **le compte std peut remonter au répertoire parent, jusqu'à la racine du serveur ftp**, ce qui peut être très dommageable au niveau de la sécurité.



Enfermer les utilisateurs anonymes

Dans le fichier `proftpd.conf` nous remarquons une balise `anonymous` commentée par défaut, nous l'avons décommentée pour autoriser l'accès aux utilisateurs anonymes.

Cette section de code configure l'accès anonyme dans un serveur ProFTPD.

En voici un résumé :

- Autorise les connexions anonymes : Les utilisateurs peuvent se connecter avec le nom

d'utilisateur "anonymous" ou "ftp".

- Fonctionnalités limitées : Les utilisateurs peuvent uniquement télécharger des fichiers (pas de téléversement).

Mesures de sécurité :

- Limite le nombre de connexions anonymes simultanées (10 dans ce cas).
- Masque la propriété réelle des fichiers pour des raisons de sécurité.
- Désactive une vérification de sécurité pour les utilisateurs anonymes (déconseillé pour une sécurité maximale).
- Messages de bienvenue : Affiche des messages lors de la connexion et du changement de répertoire.

En bref, ce code permet le téléchargement anonyme de fichiers de base à partir du serveur FTP.

Cette section est codée ainsi:

```
# A basic anonymous configuration, no upload directories.

<Anonymous ~ftpdocs>
  User ftp
  Group nogroup
  # We want clients to be able to login with "anonymous" as well as "ftp"
  UserAlias anonymous ftp
  # Cosmetic changes, all files belongs to ftp user
  DirFakeUser on ftp
  DirFakeGroup on ftp

  RequireValidShell off

  # Limit the maximum number of anonymous logins
  MaxClients 10

  # We want 'welcome.msg' displayed at login, and '.message' displayed
  # in each newly chdired directory.
  DisplayLogin welcome.msg
  DisplayChdir .message

  # Limit WRITE everywhere in the anonymous chroot
  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>

  # Uncomment this if you're brave.
  # <Directory incoming>
  #   # Umask 022 is a good standard umask to prevent new files and dirs
  #   # (second parm) from being group and world writable.
  #   Umask 022 022
  #   <Limit READ WRITE>
```

```
#      DenyAll
#      </Limit>
#      <Limit STOR>
#      AllowAll
#      </Limit>
# </Directory>
```

```
</Anonymous>
```

- **<Anonymous ~ftpdocs>** : Début de la section de configuration pour les utilisateurs anonymes. Le répertoire racine par défaut est défini comme “/ftpdocs”.
- **User ftp**: L'utilisateur interne utilisé pour les connexions anonymes est “ftp”.
- **Group nogroup**: Le groupe par défaut pour les utilisateurs anonymes est “nogroup”.
- **UserAlias anonymous ftp**: Les utilisateurs qui se connectent avec “anonymous” seront mappés à l'utilisateur interne “ftp”.
- **DirFakeUser on ftp**: Le nom d'utilisateur affiché dans les listes de répertoires sera “ftp” pour tous les fichiers.
- **DirFakeGroup on ftp**: Le groupe affiché dans les listes de répertoires sera “ftp” pour tous les fichiers.
- **RequireValidShell off**: La vérification de la présence d'un shell valide pour les utilisateurs anonymes est désactivée.
- **DisplayLogin welcome.msg**: Le message “welcome.msg” sera affiché lors de la connexion des utilisateurs anonymes.
- **DisplayChdir .message**: Le message “.message” sera affiché à chaque changement de répertoire.
- **<Directory >**: Début de la section de configuration pour tous les répertoires.
- **<Limit WRITE>**: Début de la section de configuration pour les restrictions d'écriture.
- **DenyAll**: Tout accès en écriture est interdit pour tous les utilisateurs anonymes dans tous les répertoires.
- **</Limit>**: Fin de la section de configuration pour les restrictions d'écriture.
- **</Directory>**: Fin de la section de configuration pour tous les répertoires.
- **#<Directory incoming>**: Début de la section de configuration pour le répertoire “incoming”.
- **#Umask 022 022**: Définit l'umask par défaut pour les nouveaux fichiers et répertoires dans le répertoire “incoming”.
- **#<Limit READ WRITE>**: Début de la section de configuration pour les restrictions de lecture et d'écriture.
- **#DenyAll**: Interdit tout accès en lecture et en écriture pour tous les utilisateurs dans le répertoire “incoming”.
- **#</Limit>**: Fin de la section de configuration pour les restrictions de lecture et d'écriture.
- **#<Limit STOR>**: Début de la section de configuration pour les restrictions de téléversement.
- **#AllowAll**: Autorise le téléversement pour tous les utilisateurs dans le répertoire “incoming”.
- **#</Limit>**: Fin de la section de configuration pour les restrictions de téléversement.
- **#</Directory>**: Fin de la section de configuration pour le répertoire “incoming”.
- **</Anonymous>**: Fin de la balise anonymous

Création des comptes

Présentation

Nous allons créer 3 comptes différents :

- **intra** : : **10.31.200.15** sur le port **2100** le répertoire racine sera /srv/ftp/intranet.
Permission = rw
- **extra** : **10.31.200.16** sur le port **2200** le répertoire racine sera /srv/ftp/extranet.
Permission = r
- **anonymous** : anonymous pourra accéder r qu'au répertoire /home/ftpdocs sans authentication. Permission = r

Création des comptes

pour créer un compte, il faut taper la commande adduser la création des compte s'effectue comme ceci:

```
adduser intra
```

```
adduser extra
```

```
adduser anonymous
```

Les comptes ont été créés, vérifions la création des comptes avec la commande:

```
tail -4 /etc/passwd
```

nous observons que les comptes on été bien créés.

```
root@ftp-pub:/etc/proftpd# tail -4 /etc/passwd
ftp:x:105:65534::/srv/ftp:/usr/sbin/nologin
std:x:1000:1000:::/home/std:/bin/bash
intra:x:1001:1001:::/home/intra:/bin/bash
extra:x:1002:1002:::/home/extra:/bin/bash
root@ftp-pub:/etc/proftpd#
```

Création des répertoires

Cependant les répertoire de connexion lors d'une connexion ftp avec l'un de ces comptes sera les répertoires par défaut c'est à-dire ~, nous voulons que la connexion s'effectue dans le répertoire /srv/ftp/ commençons par **créer les répertoires de connexion avec la commande mkdir**:

```
mkdir /srv/ftp/intranet
mkdir /srv/ftp/extranet
mkdir /home/ftpdocs
```

Vérifions la création des répertoire :

```
ls /srv/ftp
```

```
root@ftp-pub:/etc/proftpd# ls /srv/ftp
extranet intranet welcome.msg
root@ftp-pub:/etc/proftpd#
```

```
ls /home
```

```
root@ftp-pub:/etc/proftpd# ls /home
extra ftpdocs intra std
root@ftp-pub:/etc/proftpd#
```

Configuration des virtualhosts

Présentation

Un virtualhost, ou hébergeur virtuel, permet à un serveur web d'héberger plusieurs sites internet sur une même machine physique, parfois même avec une seule adresse IP. C'est comme si on divisait le serveur en plusieurs parties virtuelles, chacune hébergeant un site web distinct. C'est une technique courante utilisée par les hébergements mutualisés, qui permettent à de nombreux clients de partager les ressources d'un serveur unique. Grâce aux virtualhosts, chaque client peut avoir son propre site web avec son nom de domaine, tout en partageant l'infrastructure matérielle du serveur.

Configuration des virtualhost

Pour ajouter un serveur virtuel dans proftpd il faut configurer le fichier `virtuals.conf` situé dans le répertoire `/etc/proftpd/`; tapons la commande suivante :

```
nano /etc/proftpd/virtuals.conf
```

Le virtualhost d'intra est configuré comme ceci:

```
<VirtualHost 10.31.200.15>
    Port                2100
    ServerAdmin          iyansagnediagne@beaupeyrat.fr
    ServerName           "FTP Intranet"
    User                 intra
    Umask                022

    <Limit LOGIN>
        Order Allow, Deny
        Allowgroup intra
        Deny from all
    </Limit>
```



```
TransferLog           /var/log/proftpd/xfer/intranet.gsb.org
MaxLoginAttempts      3
RequireValidShell     no
DefaultRoot           /srv/ftp/intranet
AllowOverwrite        yes
</VirtualHost>
```

Et le virtualhost d'extra est configuré comme ceci:

```
<VirtualHost 10.31.200.16>
  Port                2200
  ServerAdmin         iyansagnediagne@beaupeyrat.fr
  ServerName          "FTP Extranet"
  User                extra
  Umask               022

  <Limit LOGIN>
    Order Allow, Deny
    Allowgroup intra
    Deny from all
  </Limit>

  TransferLog         /var/log/proftpd/xfer/extranet.gsb.org
  MaxLoginAttempts    3
  RequireValidShell   no
  DefaultRoot         /srv/ftp/extranet
  AllowOverwrite      yes
</VirtualHost>
```

Permissions

Pour que les connexions FTP des utilisateurs fonctionnent, **ils doivent être propriétaire de de leurs repertoire.**

Std doit pouvoir lire et écrire dans son repertoire pour cela il faut lui donner la propriété du repertoire /home/std utilisons la commande **chown** et **chmod** pour qu'il puisse lire et écrire dans son repertoire. Tapons ces commande :

```
chown -R std /home/std
chmod -R 700 /home/std
```

intra doit être propriétaire du repertoire /srv/ftp/intranet/ et doit pouvoir lire r et écrire w dans son repertoire personnel:

```
chown -R intra /srv/ftp/intranet
chmod -R 700 /srv/ftp/intranet
```

extra doit être propriétaire du répertoire /srv/ftp/extranet/ et doit pouvoir seulement lire r dans son répertoire personnel:

```
chown -R extra /srv/ftp/extranet
chmod -R 500 /srv/ftp/extranet
```

anonymous doit être propriétaire du répertoire /home/ftpdocs/ et doit pouvoir seulement lire r dans son répertoire ftpdocs:

```
chown -R anonymous /home/ftpdocs
chmod -R 500 /home/ftpdocs
```

From:

<https://sisr2.beaupeyrat.com/> - Documentations SIO2 option SISR

Permanent link:

<https://sisr2.beaupeyrat.com/doku.php?id=sisr2-europe:configuration>

Last update: **2024/12/16 14:11**

