CHAPITRE 3

FTP

Commençons par installer le serveur FTP :

```
ALBAN@bebeserv:~$ sudo -i
Password:
bebeserv:~# apt-get update
[...]
bebeserv:~# apt-get install vsftpd libdb3-util ftp
[...]
```

L'installateur créé automatiquement un utilisateur « ftp » qui à le groupe « nogroup » et le dossier « /home/ftp » pour cet utilisateur.

Nous allons changer tout ça, et de plus vu que de base « vsftp » autorise les connexions anonymes par défaut, même si « iptables » ne laissera rien passer, c'est un bon reflex de couper le serveur en attendant sa configuration finale.

```
bebeserv:~# /etc/init.d/vsftpd stop
[...]
```

Nous allons d'abord nettoyer un peu ce que « vsftpd » à créer automatiquement

```
bebeserv:~# userdel ftp
bebeserv:~# rmdir /home/ftp
```

Je vais conserver le groupe « nogroup » et utiliser l'utilisateur « nobody » qui existe déjà. Nous devons juste accorder cet utilisateur pour ce que nous voulons en faire. Ici j'anticipe un peu sachant que je vais créer des utilisateurs FTP virtuels pour mes sites internet.

```
bebeserv:~# usermod -g nogroup -d /home/nobody -s /usr/sbin/nologin
nobody
bebeserv:~# useradd -g nogroup -d /home/nobody -s /usr/sbin/nologin
anonymous
bebeserv:~# mkdir /home/nobody
bebeserv:~# chgrp nogroup /home/nobody
bebeserv:~# chmod 2770 /home/nobody
```

Créons aussi le dossier de notre premier utilisateur virtuel :

```
bebeserv:~# mkdir /home/nobody/test/
bebeserv:~# chown nobody /home/nobody/test/
```

Une petite explication s'impose, nous donnons d'abord au dossier le bon groupe et nous le mettons un « setgid » dessus afin que tous fichiers et dossiers créés dans ce dossier appartiennent au groupe « nogroup », ensuite nous ne donnons aucun droit d'écriture sur le dossier sauf au propriétaire « root » car « vsftpd » accorde automatiquement les droits d'écriture dans le dossier en fonction de ces droits et nous ne souhaitons pas forcément que tous les utilisateurs puisse écrire dedans. Nous gérerons ces droits dans « vsftpd ».

Maintenant nous allons configurer notre serveur FTP.

```
bebeserv:~# nano /etc/vsftpd.conf
```

Je change les lignes suivantes :

```
anonymous_enable=NO
local_enable=YES
write_enable=NO
local_umask=022
anon_upload_enable=NO
anon_mkdir_write_enable=NO
dirmessage_enable=NO
chown_uploads=NO
chown_username=nobody
xferlog_file=/var/log/vsftpd.log
xferlog_std_format=YES
idle_session_timeout=300
data_connection_timeout=60
nopriv_user=anonymous
ftpd_banner=Welcome to bebeserv.bebenet.local FTP service.
chroot_local_user=YES
```

Et je rajoute celles-ci :

```
guest_enable=YES
guest_username=nobody
max_clients=20
max_per_ip=5
force_dot_files=YES
hide_ids=YES
use_localtime=YES
user_config_dir=/etc/vsftpd/user_conf

# Connection sécurisée
#ssl_enable=YES
```

```
#allow_anon_ssl=YES
#force_local_data_ssl=YES
#force_local_logins_ssl=YES
#ssl_sslv2=YES
#ssl_sslv3=YES
#ssl_tlsv1=YES
```

Enregistrez et quittez.

Maintenant configurons le fichier d'authentification PAM de « vsftpd » afin d'utiliser des utilisateurs virtuels :

```
bebeserv:~# nano /etc/pam.d/vsftpd
```

Commentez toutes les lignes et mettez ceci à la fin du fichier :

```
auth required    /lib/security/pam_userdb.so db=/etc/vsftpd/login
account    required    /lib/security/pam_userdb.so
db=/etc/vsftpd/login
```

Enregistrez et quittez.

Maintenant allons créer notre base d'utilisateurs virtuels :

```
bebeserv:~# mkdir /etc/vsftpd
bebeserv:~# nano /etc/vsftpd/userdb
```

Créez l'utilisateur « test » correspondant au dossier que nous avons fait plus haut :

```
test
toto

```

Vérifiez toujours de bien mettre une ligne vide en fin de fichier !

La seconde ligne est le mot de passe de l'utilisateur ici donc « toto ». Le fichier doit être écrit strictement de la sorte (avec toujours une ligne vide à la fin) :

```
user1
pass1
user2
pass2
user3
pass3
```

```
...
```

Enregistrez et quittez.

Maintenant formatons notre base de données au format « userdb ». pour ceci plutôt que de taper une ligne de commande fastidieuse et dont on ne se rappellera jamais, nous allons créer un petit script qui le fera pour nous.

Créons notre script :

```
bebeserv:~# nano vsftpd-makedb
```

Écrivez les lignes suivantes :

```
#!/bin/sh
db3_load -T -t hash -f /etc/vsftpd/userdb /etc/vsftpd/login.db
```

Enregistrez et quittez.

Maintenant donnons les bons droits à notre script :

```
bebeserv:~# chmod 500 vsftpd-makedb
```

Ajoutons un lien symbolique afin d'avoir accès directement à cette nouvelle commande, comme toutes les autres commandes.

```
bebeserv:~# ln -s /etc/vsftpd/vsftpd-makedb /usr/local/bin/
```

Maintenant et après chaque modification de la table « userdb » il faudra faire :

```
bebeserv:~# vsftpd-makedb
```

Dorénavant notre utilisateur « test » existe pour « vsftpd » donnons lui des droits afin que « vsftp » sache ce qui lui est autorisé. Mais pour ceci, afin de nous rappeler des options en permanence, nous allons créer d'abord un fichier d'exemple :

```
bebeserv:~# mkdir /etc/vsftpd/user_conf
bebeserv:~# cd /etc/vsftpd/user_conf
```

```
bebeserv:/etc/vsftpd/user_conf# nano example
```

Écrivez :

```
# donne les droits de lecture
download_enable=YES
anon_world_readable_only=NO

# donne les droits d'écriture
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES

# donne le droit de renommer de supprimer
anon_other_write_enable=YES

#donne le droit de faire des chmod
chmod_enable=YES
virtual_use_local_privs=YES

#affiche les fichier cachés
force_dot_files=YES

# donne un home dans /home/nobody
# exemple : local_root=test le home de l'utilisateur sera :
# /home/nobody/test
local_root=test

# droits des fichiers créés
anon_umask=002
```

Enregistrez et quittez.

Maintenant que nous avons un fichier d'exemple, créez celui de l'utilisateur « test » et modifiez les options si vous le souhaiter :

```
bebeserv:/etc/vsftpd/user_conf# cp example test
```

Si ce n'est déjà fait ajoutez les lignes suivantes à votre fichier « /etc/network/iptables » :

```
iptables -t filter -A INPUT -i eth0 -p tcp --dport 21 -d $MYIP -j
ACCEPT
iptables -t filter -A INPUT -i eth0 -p tcp --sport 20 -d $MYIP -j
ACCEPT

iptables -t filter -A OUTPUT -o eth0 -p tcp --dport 21 -s $MYIP -j
ACCEPT
iptables -t filter -A OUTPUT -o eth0 -p tcp --sport 20 -s $MYIP -j
ACCEPT
```

Et à la fin du fichier rajoutez :

```
modprobe ip_conntrack_ftp ports=21
```

Configurons « fail2ban » pour qu'il scanne les connexions FTP :

```
bebeserv:/etc/vsftpd/user_conf# nano /etc/fail2ban/jail.conf
```

Modifiez cette ligne :

```
[vsftpd]
enabled  = true
```

Enregistrez et quittez.

Maintenant que tout est prêt, redémarrez « iptables » et « vsftpd » :

```
bebeserv:/etc/vsftpd/user_conf# /etc/init.d/vsftpd start
[...]
bebeserv:/etc/vsftpd/user_conf# /etc/init.d/iptables-conf restart
[...]
```

Faisons un petit test de connexion :

```
bebeserv:/etc/vsftpd/user_conf# ftp localhost 21
Connected to localhost.
220 Welcome to bebeserv.bebenet.local FTP service.
Name :
```

Loguez vous avec l'utilisateur « test » avec le mot de passe « toto ».

La connexion réussi votre serveur ftp fonctionne.

Quittez :

```
ftp> quit
221 Goodbye.
bebeserv:/etc/vsftpd/user_conf#
```

Dans cette configuration néanmoins les utilisateurs locaux ne peuvent pas se connecter en FTP classique. ce qui est en même temps un bonne chose car le FTP classique peut être remplacer

par du SFTP (FTP par SSH) vous n'avez rien à changer pour que ceci fonctionne.

ATTENTION ne confondez pas SFTP ET FTPS (FTP par SSL).

Afin de sécurisé aussi nos comptes virtuels nous allons faire en sorte qu'ils puisse se connecter en FTPS.

Pour ceci nous allons avoir besoin de « open-ssl » pour générer les certificats adéquats et la connexion SSL.

```
bebeserv:/etc/vsftpd/user_conf# apt-get install openssl
[...]
```

Maintenant générons un certificat pour « vsftpd » :

```
bebeserv:/etc/vsftpd/user_conf# cd /etc/ssl/certs
bebeserv:/etc/ssl/certs# openssl req -x509 -nodes -days 730 -newkey
rsa:1024 -keyout vsftpd.pem -out vsftpd.pem
```

Aux questions répondez comme ceci par exemple; le plus important étant le « Common Name » qui doit être l'adresse IP exact ou le nom de domaine exact du service), sachant que nous configurerons plus tard un sous domaine nommé « ftp » sur notre domaine « bebenet.local » j'utiliserai celui çi : « ftp.bebenet.local ».

Voici ce que j'ai répondu :

```
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:France
Locality Name (eg, city) []:Paris
Organization Name (eg, company) [Internet Widgits Pty
Ltd]:bebenet.local
Organizational Unit Name (eg, section) []:bebenet.local
Common Name (eg, YOUR name) []:ftp.bebenet.local
Email Address []:alban@bebenet.local
```

Vous venez de créer un certificat valide pendant deux ans.

Maintenant sécurisez votre certificat avec des droits correct :

```
bebeserv:/etc/ssl/certs# chmod 0600 vsftpd.pem
```

Maintenant décommentez les lignes suivantes dans « /etc/vsftpd.conf » :

```
# Connection sécurisée
ssl_enable=YES
allow_anon_ssl=YES
force_local_data_ssl=YES
force_local_logins_ssl=YES
ssl_sslv2=YES
ssl_sslv3=YES
ssl_tlsv1=YES
```

Enregistrez et quittez.

Rechargez la configuration de « vsftpd » :

```
bebeserv:/etc/ssl/certs# /etc/init.d/vsftpd reload
[...]
```

Vous pouvez régler le client de votre utilisateur virtuel pour se connecter en FTPS (FTP with SSL auth SSL explicit) ou (FTP with SSL auth TLS explicit).

Pensez à régler votre client en mode actif sans quoi il n'ira pas plus loin que la connexion. En effet sans SSL le serveur est régler pour géré le mode actif comme passif sans problème, seulement SSL empêche la gestion du mode passif. Pour contrer ceci nous pourrions limité la plage de ports passif sous « vsftpd » et ouvrir dans « iptables » ces ports mais cette idée je ne l'implémenterai pas ici.

Un petit rappel, une fois le SSL activer, les clients ne pourront plus se connecter sans mode SSL. pour les utilisateurs locaux SFTP (SSH) fonctionne toujours correctement.

Voilà pour ce chapitre.

```
bebeserv:/etc/ssl/certs# exit
logout
ALBAN@bebeserv:~$
```