VsFTPD + Mysql Auth

Objectif :

Ce tutoriel permettra d'authentifier des utilisateurs VsftpD grâce à une base de données MySQL. Le but étant de ne pas utiliser une base de données Berkeley qui doit être re-génénée à chaque création, suppression ou mise à jour d'utilisateur.

VsFTPD + Mysql Auth

Pré-requis

Installez les paquets si se n'est déjà pas fait
vsftpd mysql-server libpam-mysql openssl

VsFTPD + Mysql Auth

Configuration de VsftpD

Configuration de VsftpD

```
~$ sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.OK
```

VsFTPD + Mysql Auth

Éditez le fichier en tant qu'administrateur.

Ci-dessous la configuration du fichier */etc/vsftpd.conf* avec le détail de chaque option. Les utilisateurs virtuels se connectent et sont emprisonnés (chroot) dans un même dossier, les connections sont encryptées en SSL (Cipher DES-CBC3-SHA) et les options de connexion passives sont activées.

VsFTPD + Mysql Auth

```
# Serveur en ecoute
listen=YES
# Port d'ecoute du serveur
listen_port=21

# Options des utilisateurs anonymes ici désactivés
anonymous_enable=NO
# Autorisation d'upload pour les utilisateurs anonymes
anon_upload_enable=NO
# Autorisation de création de répertoire pour les utilisateurs anonymes
# mais aussi de suppression et de renommage
anon_other_write_enable=NO
anon_mkdir_write_enable=NO
# Autorisation de connexion anonyme en ssl
allow_anon_ssl=NO
# Autorise les utilisateurs locaux ou virtuels a se connecter
local_enable=YES
# Autorise l'ecriture sur le serveur (upload)
write_enable=YES
# Masque d'upload de fichier 022 => correspond a un chmod 755
local_umask=022
# Monitoring de base via `ps -ef | grep vsftpd`
setproctitle_enable=YES
# Active les messages de changement de repertoire
dirmessage_enable=YES
# Utilisation de log pour les uploads et downloads (par defaut /var/log/vsftpd.log)
xferlog_enable=YES
# Emplacement du fichier de log
xferlog_file=/var/log/vsftpd.log
# Formatage de la log au standard wu-ftpd
xferlog_std_format=YES
# Utilisation de 2 fichiers de log differents (Par defaut /var/log/xferlog et /var/log/vsftpd.log)
dual_log_enable=YES
```

VsFTPD + Mysql Auth

```
# Options de connexion
# Nombre de clients maximum
max_clients=30
# Nombre maximum de connections par clients
max_per_ip=3
# Duree en secondes d'inactivite avant deconnexion de la session
idle_session_timeout=60
# Duree en secondes d'inactivite avant deconnexion de donnees
data_connection_timeout=120
# Debit maximum du serveur en bytes par secondes (0 = debit illimite)
local_max_rate=0
# Message de bienvenue affiche durant la phase de connexion
ftpd_banner=Bienvenue sur le serveur VsftpD
# Bloquer les utilisateurs dans un dossier
chroot_local_user=YES
# Dossier utiliser pour le chroot (doit appartenir a root et avec un chmod 755)
# car il ne doit surtout pas etre inscriptible (writable) par tous le monde
secure_chroot_dir=/var/run/vsftpd
# Nom du service d'authentification utilise par le serveur vsftpd
pam_service_name=vsftpd
# Utilisation des privileges locaux pour les utilisateurs virtuels
# permet notamment de donner les droits d'ecriture car sinon les
# utilisateurs virtuels ont des droits d'utilisateurs anonymes
virtual_use_local_privs=YES
# Autoriser les utilisateurs virtuels
guest_enable=YES
# Utilisateur du lancement du serveur vsftpd
# ici c'est le meme utilisateur que le serveur apache
# mais vous pouvez creer un utilisateur dedie a cette tache
guest_username=www-data
# Dossier ou vont etre encapsules les utilisateurs virtuels
# represente le / du site ftp
local_root=/home/vsftpd
```

VsFTPD + Mysql Auth

Options supplémentaires pour le SSL, utiles si vous voulez encrypter les connexions utilisateur et connexions de données.

```
# Activation du SSL
ssl_enable=YES
# Oblige les connexions de donnees a passer par du SSL
# Si cette option est activee les clients ftp ne gerant pas
# SSL ne pourront envoyer ni recevoir de donnees
force_local_data_ssl=NO
# Oblige la connexion d'identification a etre encryptee en SSL
# Si cette option est activee les clients ftp ne gerant pas
# SSL ne pourront plus se connecter
force_local_logins_ssl=YES
# Versions de SSL pris en charge par le serveur VsftpD
ssl_sslv2=YES
ssl_sslv3=YES
ssl_tlsv1=YES
# Emplacement du certificat d'encryption SSL
rsa_cert_file=/etc/ssl/certs/vsftpd/vsftpd.pem
# Emplacement de la clé privée (inutile et ne fonctionne pas ,il va la chercher dans
le cert)
#rsa_private_key_file=/etc/ssl/certs/vsftpd/vsftpd.key
```

VsFTPD + Mysql Auth

Dossier partagé

Il faut penser à créer le dossier **secure_chroot_dir Vsftpd** (dans notre exemple */var/run/vsftpd*) et lui appliquer les permissions adéquates, il ne faut pas que les utilisateurs aient des droits d'écriture dans ce répertoire.

```
sudo mkdir /var/run/vsftpd
sudo chown root:root /var/run/vsftpd
sudo chmod 660 /var/run/vsftpd
```

VsFTPD + Mysql Auth

Configuration de MySQL

Cette partie nous concerne peu car ayant déjà la base d'authentification centralisée dans tous les CNF. Cependant je la met juste à titre d'exemple.

À ce stade, plusieurs possibilités s'offrent pour administrer la base de données MySQL :

En commande, pas très convivial mais efficace ou via une interface Web comme PhpMyAdmin ou Webmin.

VsFTPD + Mysql Auth

Configuration de MySQL

```
sudo mysql -u root -p (puis taper le mot de passe de votre superu-
tilisateur MySQL)
mysql>

# Création d'une nouvelle base nommée "vsftpd"
CREATE DATABASE vsftpd;
# Attribution des privileges a l'utilisateur vsftpd avec comme mot
de passe : MOTDEPASSE_VSFTPD
# MOTDEPASSE_VSFTPD ne doit pas contenir de caractère # : inter-
préte le reste de la ligne comme un commentaire.
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP ON vsftpd.* TO
'vsftpd'@'localhost' IDENTIFIED BY 'MOTDEPASSE_VSFTPD';
# Application des privileges
FLUSH PRIVILEGES;
# Utilisation de la base de donnees fraichement creee
USE vsftpd;
```

VsFTPD + Mysql Auth

```
# Creation d'une table utilsateurs avec 4 champs (ID, NOM, PASS,
CRYPTAGE)
# ID => identifiant unique (auto-incrementation et cle primaire)
# NOM => texte (nul non autorise)
#
# PASS => texte (nul non autorise)
# CRYPTAGE => texte (nul autorise)
CREATE TABLE `utilisateurs` ( `ID` INT NOT NULL AUTO_INCREMENT
PRIMARY KEY , `NOM` TEXT NOT NULL , `PASS` TEXT NOT NULL , `CRYP-
TAGE` TEXT );
# Creation d'une table logging avec 6 champs (ID, USER, HOST,
RHOST, MSG, TIME)
# ID => identifiant unique (auto-incrementation et cle primaire)
# USER => texte (nul non autorise)
# HOST => texte (nul non autorise)
# RHOST => texte (nul non autorise)
# TIME => texte (nul non autorise)
# MSG => => texte (nul non autorise)
CREATE TABLE `logging` ( `ID` INT NOT NULL AUTO_INCREMENT PRIMARY
KEY , `USER` TEXT NOT NULL , `HOST` TEXT NOT NULL , `RHOST` TEXT
NOT NULL , `TIME` TEXT NOT NULL , `MSG` TEXT NOT NULL );
```

VsFTPD + Mysql Auth

Configuration de MySQL

Notre base de données est créée et nous allons donc la renseigner avec un jeu d'utilisateurs de test, voici la syntaxe pour ajouter un utilisateur, sachant

Nous allons créer 3 utilisateurs nommés respectivement "toto", "tata" et "titi" avec comme mots de passes respectifs toto, tata et titi. Ceci afin de détailler l'utilisation de la librairie pam_mysql et de MySQL-server.

VsFTPD + Mysql Auth

```
# Creation de l'utilisateur toto avec comme mot de passe
toto stocke en clair (aucun cryptage)
INSERT INTO utilisateurs (NOM, PASS, CRYPTAGE)
VALUES('toto', 'toto', 'aucun' );

# Creation de l'utilisateur tata avec comme mot de passe
tata crypte avec la fonction MySQL PASSWORD() => decon-
seille dans le manuel MySQL
INSERT INTO utilisateurs (NOM, PASS, CRYPTAGE)
VALUES('tata', PASSWORD('tata'), 'PASSWORD' );

# Creation de l'utilisateur titi avec comme mot de passe
titi crypte avec la fonction MySQL ENCRYPT() => conseille
a la place de PASSWORD()
INSERT INTO utilisateurs (NOM, PASS, CRYPTAGE)
VALUES('titi', ENCRYPT('titi'), 'ENCRYPT' );
```

VsFTPD + Mysql Auth

Configuration du certificat SSL

```
# Creation du repertoire pour stocker les certificats
sudo mkdir /etc/ssl/certs/vsftpd && cd /etc/ssl/certs/vsftpd
# Creation du certificat SSL valable 1 an
sudo openssl req -x509 -nodes -days 365 -newkey rsa:1024
-keyout vsftpd.pem -out vsftpd.pem
# Copie de la partie [PRIVATE_KEY] du certificat dans le
fichier vsftpd.key
sudo head -15 vsftpd.pem > vsftpd.key
# Protection du certificat et de la cle privee
sudo chmod 600 *
```

La clé privée générée est auto-signée et certains clients vont donc afficher des alertes de sécurité, car la clé n'est pas validée par une authorité de certification reconnue (Thawte, Verisign, etc)

VsFTPD + Mysql Auth

Activation de la connexion via SSL

Options supplémentaires pour le SSL, utiles si vous voulez encrypter les connexions utilisateur et connexions de données.

```
# Activation du SSL
ssl_enable=YES
# Oblige les connexions de donnees a passer par du SSL
# Si cette option est activee les clients ftp ne gerant pas
# SSL ne pourront envoyer ni recevoir de donnees
force_local_data_ssl=NO
# Oblige la connexion d'identification a etre encryptee en SSL
# Si cette option est activee les clients ftp ne gerant pas
# SSL ne pourront plus se connecter
force_local_logins_ssl=YES
# Versions de SSL pris en charge par le serveur VsftpD
ssl_sslv2=YES
ssl_sslv3=YES
ssl_tlsv1=YES
# Emplacement du certificat d'encryption SSL
rsa_cert_file=/etc/ssl/certs/vsftpd/vsftpd.pem
# Emplacement de la clé privée (inutile et ne fonctionne
pas ,il va la chercher dans le cert)
#rsa_private_key_file=/etc/ssl/certs/vsftpd/vsftpd.key
```

VsFTPD + Mysql Auth

Configuration de pam_mysql

Éditez le fichier /etc/pam.d/vsftpd vous devriez tomber sur quelque chose ressemblant à ça :

```
# Standard behaviour for ftpd(8).
auth    required        pam_listfile.so item=user sense=deny
file=/etc/ftpusers onerr=succeed

# Note: vsftpd handles anonymous logins on its own.  Do not
enable
# pam_ftp.so.

# Standard blurb.
@include common-account
@include common-session

@include common-auth
auth    required        pam_shells.so
```

VsFTPD + Mysql Auth

Il faut commenter toutes les lignes avec un # ou effacer tout le contenu du fichier et coller le code ci dessous à la place.

```
# fonction pam_mysql crypt=1 OK avec la fonction ENCRYPT() de MySQL
# fonction pam_mysql crypt=2 OK avec la fonction PASSWORD() de MySQL

# Minimum necesaire afin de se connecter
# auth required pam_mysql.so user=vsftpd passwd=VsftpD host=localhost db=vsftpd table=users usercolumn=nom passwdcolumn=mdp crypt=1
# account required pam_mysql.so user=vsftpd passwd=VsftpD host=localhost db=vsftpd table=users usercolumn=nom passwdcolumn=mdp crypt=1

# Connexion avec logging en base de donnees des acces
auth required pam_mysql.so verbose=1 user=vsftpd passwd=VsftpD host=localhost db=vsftpd table=utilisateurs usercolumn=NOM passwdcolumn=PASS crypt=1 sqllog=true logtable=logging logmsgcolumn=msg logusercolumn=user loghostcolumn=host logrhostcolumn=rhost logtimecolumn=time
account required pam_mysql.so verbose=1 user=vsftpd passwd=VsftpD host=localhost db=vsftpd table=utilisateurs usercolumn=NOM passwdcolumn=PASS crypt=1 sqllog=true logtable=logging logmsgcolumn=msg logusercolumn=user loghostcolumn=host logrhostcolumn=rhost logtimecolumn=time
```

VsFTPD + Mysql Auth

A ne pas oublier de remplacer VsftpD par le mot de passe choisi lors de la création de la base de données

VsFTPD + Mysql Auth

La librairie pam_mysql acceptent plusieurs arguments dont voici le détail :
verbose ⇒ Mode verbeux, nécessaire pour logger les accès (0=désactivé, 1 = activé)
user ⇒ Utilisateur employé par VsftpD pour se conncter à MySQL
password ⇒ Mot de passe de l'utilsateur employé par VsftpD pour se connecter à MySQL
host ⇒ Hote hébergeant le serveur MySQL (localhost ou adresse IP)
db ⇒ nom de la base de données à utiliser
table ⇒ nom de la table contenant les utilisateurs
usercolumn ⇒ nom de la colonne contenant les noms des utilisateurs
passwdcolumn ⇒ nom de la colone contenant les mot de passe des utilisateurs
crypt ⇒ type de cryptage utilisé pour les mots de passe (0 = clair , 1 = fonction ENCRYPT(), 2 = fonction PASSWORD(), 3 = fonction MD5(), 4 = fonction SHA1())

VsFTPD + Mysql Auth

sqllog ⇒ activation du logging d'accès en base SQL (0 = désactivé, 1 = activé)
logtable ⇒ nom de la table de log des accès utilisateurs
logmsgcolumn ⇒ nom de la colonne ou seront stockés les messages de pam_mysql
logusercolumn ⇒ nom de la colonne ou seront stockés les nom des utilisateurs
logpidcolumn ⇒ nom de la colonne ou seront stockés les numéros de process (pid)
loghostcolumn ⇒ nom de la colonne ou seront stockés les adresses ou se connectent les utilisateurs (en général le serveur lui même)
logrhostcolumn ⇒ nom de la colonne ou seront stockés les adresses distantes des utilisateurs
logtimecolumn ⇒ nom de la colonne ou seront stockés les heures de connexion

VsFTPD + Mysql Auth

La fonction **crypt** de la librairie pam_mysql accepte différents arguments (0, 1, 2, 3, 4) , cependant je n'en ai trouvé que deux qui fonctionnent avec mysql-server, bien que, sous mysql la fonction MD5() et la fonction SHA1() existent, elles ne renvoient pas les mêmes valeurs que le crypt fourni par pam_mysql. Autrement dit, les valeurs ne correspondent pas et l'authentification échoue systématiquement.

VsFTPD + Mysql Auth

Utilisation

Pour faire mes tests j'ai utilisé différents clients ftp, cependant je conseille ftp-ssl pour les tests car les messages d'erreurs sont plus parlants. Autrement il existe Filezilla ou FireFTP sous Mozilla qui fonctionnent très bien.

Installez les paquets suivants :

- **filezilla**
- **ftp-ssl**
- **Plugin FireFTP pour Firefox**

Pour commencer il faut recharger le fichier de configuration VsftpD.

sudo /etc/init.d/vsftpd restart

VsFTPD + Mysql Auth

Ensuite tentative de connexion en commande ou via un client graphique qui devrait normalement vous répondre comme ci-dessous. Il est aussi possible de tester via un navigateur à l'adresse ftp://toto@localhost:21 si vous avez laisser les options adéquates.

VsFTPD + Mysql Auth

```
ftp-ssl localhost 21
Connected to localhost.
220 Bienvenue sur le serveur VsftpD
Name (localhost:toto): toto
234 Proceed with negotiation.
[SSL Cipher DES-CBC3-SHA]
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

VsFTPD + Mysql Auth

Si toutefois vous avez des problèmes de connexion (Authentication Failed) vérifiez les paramètres de cryptage de pam_mysql et la fonction utilisée dans MySQL-server pour crypter le mot de passe. Pour les problèmes de connexion sur nom de domaine vérifiez bien que tous les ports (connexion et plage de ports passifs) sont bien ouverts et redirigés vers l'adresse IP locale de votre machine.