SFTP : Accès utilisateur via openssh-server

- Objet : Créer un accès SFTP-only via OpenSsh en accès à distance
- Niveau requis :
  débutant, avisé
- Commentaires : *Réalisé sur Debian Wheezy 7.5 (kimsufi OVH)*
- Débutant, à savoir : Utiliser GNU/Linux en ligne de commande, tout commence là !.
- Suivi :
  - Création par Galliezb le 08/07/2014
  - Testé par Galliezb le 08/07/2014
- Commentaires sur le forum : ici [1)]

Introduction

Vous souhaitez bénéficier d'un accès *à la* FTP à votre serveur, configurable rapidement sans négliger la sécurité ?
Voyons comment cela est possible.

Installation

Il vous faudra le serveur ssh `openssh-server` d'installé sur le serveur.


Je peux vérifier qu'OpenSsh est installé via la commande apt-cache policy openssh

Installation d'OpenSsh-server

```
apt-get update && apt-get install openssh-server
```

C'est tout ce dont vous avez besoin. C'est génial non ?

Configuration

Création de l'utilisateur

La première chose dont nous allons avoir besoin, c'est d'un utilisateur neutre. Par neutre j'entends « *un utilisateur qui accède au FTP et à rien d'autre* »

Créons notre utilisateur.


Notre utilisateur test se nommera ici `debora95dftp`

Pour créer notre utilisateur et lui affecter un mot de passe :

```
adduser debora95dftp
```

Si le répertoire :

```
/home/debora95dftp/
```

n'est pas présent, utilisez mkdir pour le créer (*dans mon cas, c'est fait automatiquement*)

Configuration d'OpenSSH

Modifions la configuration d'openssh pour permettre à cet utilisateur de se connecter.

```
nano /etc/ssh/sshd_config
```

Voici les lignes à modifier :

Extrait de sshd_config

```
port 22
PermitRootLogin no
AllowUsers debora95dftp vos-éventuels-autres-utilisateurs
[...]
UsePAM yes
[...]
Match user debora95dftp
        ChrootDirectory /home/debora95dftp/
        ForceCommand internal-sftp
        AllowTCPForwarding no
```

Le match user doit se trouver après le UsePAM. Dans le cas contraire, vous recevrez une erreur lors de la relance du service ssh.

Explications :

- **`Port 22`** : la valeur d'écoute du service ssh, pensez à changer de port, c'est mieux pour la sécurité.
- **`PermitRootLogin no`** : on ne se connecte pas en root sur le serveur. On passe par un utilisateur qui lui passera en root. Toujours mieux pour la sécurité.
- **`AllowUsers debora95dftp vos-éventuels-autres-utilisateurs`** : ici on autorise uniquement debora (et nos éventuels autres utilisateurs) à se connecter via ssh
- **`Match user debora95dftp`** : tout ce qui va suivre ne concerne que debora95dftp
- **`ChrootDirectory /home/debora95dftp/`**: debora sera cloisonnée dans son répertoire et

ne pourra pas remonter aux répertoires parents.

- **`ForceCommand internal-sftp`**: (doc infos)
- **`AllowTCPForwarding no`** : voir lien ci-dessus

ATTENTION ! : Si vous ne disposez pas d'un second utilisateur, vous n'aurez plus d'accès SSH à votre machine distante.
En effet, par la suite debora n'aura plus accès au shell, et avec l'accès root coupé, vous allez faire comment ?

Soyez sûr d'avoir configuré **un autre utilisateur** pouvant se connecter en SSH avant de continuer, ou laissez **PermitRootLogin yes** (*ce qui est vivement déconseillé*)

On redémarre ssh

```
service ssh restart
```

Notre `debora95dftp` est désormais capable de se connecter via ssh à notre serveur. Mais bizarrement, la connexion SFTP échoue.
C'est à cause du chroot qui a besoin des droits sur le répertoire en root. Nous allons donc les lui accorder ainsi :

```
chown root:root /home/debora95dftp/
```

Debora est super heureuse qu'on lui donne accès.
Elle le serait encore plus si elle pouvait écrire et supprimer dans son répertoire.
Argh, en voila un problème !
Nous allons donc donner à Debora un répertoire tout à elle, et lui accorder ses droits :

```
mkdir /home/debora95dftp/writeable/
```

```
chown debora95dftp:debora95dftp /home/debora95dftp/writeable/
```

```
chmod -R 755 /home/debora95dftp/writeable/
```

Maintenant Debora à accès à son répertoire et peut y mettre ce qu'elle veut et en faire ce qu'elle veut (*comme avec ses cheveux...*)

Aujourd'hui tout va bien, on s'aime, c'est le bonheur.
Et si demain elle apprenait que j'ai caché sur le serveur les noms et adresses de mes nombreuses maîtresses ?
Etant parano, je ne veux pas lui laisser la moindre chance de farfouiller avec sa connexion SSH car, n'étant pas un expert en administration linux, je ne dois pas lui laisser d'accès à ce qui ne lui est pas nécessaire !

Couper l'accès au shell par Debora :

```
chsh debora95dftp -s /bin/true
```

Cette commande m'as été fournie par un *girafe* (captnfab), elle fermera le shell dès que Debora se connectera en SSH.
Autrement dit, Debora n'aura accès à rien qui ne la concerne pas. (*Elle pourra prendre la maison, les gosses, la voiture mais pas mon serveur !*)

Erreur durant l'apprentissage

Pas moyen de se connecter au SFTP

- Vérifier qu'iptables ne bloque pas le port que vous souhaitez utiliser
- Vérifier que votre utilisateur est bien dans `AllowUsers` (dans `/etc/ssh/sshd_config`)
- Vérifier que le répertoire de l'utilisateur est bien en `root`

```
ls -l /home/
```

La ligne de debora95dftp doit notamment afficher :

```
 root   root
```

Vous pouvez commentez les lignes à partir du *`Match user debora95dftp`* et redémarrer le service SSH pour savoir s'il s'agit d'un problème de droit d'accès au répertoire.

Conclusion

Voici une façon simple et rapide de créer un accès SFTP qui vaut toujours mieux qu'un accès FTP (les mots de passe ne transitent pas en clair sur le réseau, et cela fait toujours un service de moins d'installé).

Dans le cas ou vous souhaitez gérer davantage d'utilisateurs, visitez cette page.

En créant des groupes et en modifiant la configuration de sshd, vous pouvez gérer le tout assez simplement.

Bon à savoir

Avant de devenir un beau chêne, on a tous été un gland.

[1)]
N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !