Serveur SSH

sécurité, SSH, système

Configuration du serveur

Editez le fichier de configuration :

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

Port par défaut

Le port par défaut est le port 22, mais il est possible de le changer (par exemple pour limiter les attaques en “brute force”)

```
Port 2222
```

Interdire la connexion de l'utilisateur root

Mettre PermitRootLogin à “no” :

```
PermitRootLogin no
```

Sécurité

```
MaxStartups 10:30:60
ClientAliveCountMax 6 # Try to login 6 times
ClientAliveInterval 20 # Try to login with intervals of 20 seconds
AllowUsers username
```

Activer l'authentification par clés SSH

Afin de sécuriser un peu plus le serveur, nous allons mettre en place le système d'authentification par clés SSH. Ainsi, seuls les utilisateurs possédant une clé privée déclarée sur le serveur pourront se connecter.

Générer une clé

Nous allons ici générer une clé RSA :

```
ssh-keygen -t rsa -b 4096 -f username
```

Deux clés sont créées dans le répertoire ~/.ssh/, username et username.pub.

Copier la clé publique sur le serveur

```
ssh-copy-id -i ~/.ssh/username.pub user@mon_serveur
```

Le contenu de la clé username sera copié dans le fichier ~/.ssh/authorized_keys

Utiliser les clés

Editez le fichier de configuration :

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

Décommentez AuthorizedKeysFile :

```
AuthorizedKeysFile      %h/.ssh/authorized_keys
```

Désactiver l'authentification par mot de passe

Décommenter et mettre PasswordAuthentication à “no” :

```
PasswordAuthentication no
```

Prise en compte des paramètres

Enfin, redémarrer le serveur ssh :

```
sudo service sshd reload
```

Aller plus loin

<note tip>Pour sécuriser un peu plus votre serveur ssh, je vous conseille d'utiliser fail2ban. Le tuto. se trouve ici.</note>