openssh-server : autoriser les connexions extérieures de manière sécurisée

- Objet : Installer un serveur ssh
- Niveau requis :
  débutant
- Commentaires : *Administrer son serveur à distance, établir un tunnel sécurisé pour relayer des ports, etc...*
- Débutant, à savoir : Utiliser GNU/Linux en ligne de commande, tout commence là ! 😀
- Suivi :
  - Création par bendia le 07/01/2014
  - Testé par captnfab le 09/02/2021
- Commentaires sur le forum : ici[1)]

Installation

Un seul paquet à installer

```
apt install openssh-server
```

Le serveur SSH est automatiquement démarré après l'installation du paquet.

Configuration

La configuration par défaut est très bonne, et il n'est pas conseillé d'y toucher à moins de bien savoir ce que vous faites.

Si vous voulez la personnaliser, il vous faut éditer le fichier `/etc/ssh/sshd_config` :

```
editor /etc/ssh/sshd_config
```

Les valeurs de configuration par défaut sont souvent présentées commentées (avec un # en début de ligne).

Comme d'hab', plus d'infos dans le :

```
man sshd_config
```

Une fois configuré il vous suffit de relancer SSH, à faire en root :

```
service ssh restart
```

Sécurité et bonnes pratiques

Un serveur SSH permet a priori à un utilisateur extérieur de se connecter sur le système. Il y a deux méthodes pour s'authentifier auprès d'un serveur SSH

- avec le nom d'utilisateur et le mot de passe d'un compte sur la machine.
- avec le nom d'utilisateur et une paire de clefs privée/publique associée à un compte sur la machine

La deuxième méthode est plus sûre que la première.


Il est important que n'importe qui ne puisse pas se connecter à votre système !

Le serveur SSH écoute sur le port 22 du système, qui est accessible depuis le réseau local.

- Si ce port n'est pas rendu accessible depuis l'extérieur, vous êtes tranquille car seuls les utilisateurs de votre réseaux peuvent essayer de se connecter (mais vous ne pouvez pas joindre votre machine depuis l'extérieur...).
- Si au contraire vous avez redirigé un port de votre box vers le port SSH de votre système, alors attention. Il y a des petits malins qui forcent le passage pour entrer chez vous en essayant des tetra...floppées de mots de passe et noms d'utilisateurs courants ;). Dans ce dernier cas, vérifiez bien qu'aucun compte n'a de mot de passe simple, ou mieux encore, bloquez la possibilité de se connecter au serveur avec un mot de passe (voir plus bas).

Il peut être intéressant de relire les conseils généraux en matière de sécurité !

Voici quelques options avec lesquelles il ne faut pas faire de blague.

Connexion par root

Il ne faut absolument pas autoriser root à se connecter par mot de passe, sauf à s'assurer d'avoir un mot de passe en béton armé[2)].

Vous pouvez autoriser les connexions à root par paire de clefs publique/privée, c'est le comportement par défaut: `PermitRootLogin prohibit-password` ou complètement bloquer toutes les connexions: `PermitRootLogin no`

Connexion par clefs

Par défaut, la connexion par clefs est autorisée, c'est très bien et reste sans conséquence si vous n'utilisez pas cette possibilité. `PubkeyAuthentication yes`

Les clefs ssh des utilisateurs sont lues par défaut dans deux fichiers du dossier personnel des utilisateurs : `~/.ssh/authorized_keys` et `~/.ssh/authorized_keys2` (déconseillé). Vous pouvez forcer l'utilisation unique du premier fichier avec `AuthorizedKeysFile .ssh/authorized_keys`.

Connexion par mot de passe

Si vous voulez interdire la connexion par mot de passe (attention à être sûr de bien pouvoir vous connecter par clef avant...), vous pouvez désactiver cette possibilité avec `PasswordAuthentication no` et `UsePAM no`

Vérifiez que vous n'autorisez pas la connexion aux comptes dont le mot de passe est vide: `PermitEmptyPasswords no`

Pour le bloquage de la connexion par mot de passe :

Dans un premier temps, laisser **“# PasswordAuthentication yes”** pour permettre la première connexion depuis ssh client, c'est-à-dire pour permettre la connexion par nom d'utilisateur et mot de passe lorsque aucune clé n'a été générée.

Puis lorsque la génération des clés asymétriques a été faite et que celles-ci ont été exportées et sont opérationnelles (la “passphrase” uniquement est demandée pour la connexion du client ssh au serveur ssh), on peut alors éditer à nouveau **/etc/ssh/sshd_config** pour mettre **`PasswordAuthentication no`**

Restreindre l'accès à quelques utilisateurs

Plutôt que de laisser possible la connexion à n'importe quel compte utilisateur (à condition d'avoir la clef ou le mot de passe), vous pouvez n'autoriser la connexion que pour certains comptes en spécifiant les noms d'utilisateur:

```
AllowUsers utilisateur1 utilisateur2
```

Ou inversement, en interdisant spécifiquement la connexion à certains comptes: `DenyUsers test guest admin root snort apache nobody`

Il est possible de spécifier plusieurs utilisateurs ou un masque d'expression régulière, je vous laisse lire la doc pour un paramétrage plus fin.

Restreindre la connexion à un groupe

Dans le même esprit que le paramètre ci-dessus on peut autoriser directement tous les utilisateurs d'un groupe.

```
AllowGroups mongroupadmin
```

Le port

Si votre serveur SSH est exposé sur internet, il peut être pratique pour limiter le bruit (et dans une

moindre mesure, augmenter la sécurité), de changer le port d'écoute de celui-ci.

On peut par exemple changer le port **22** (c'est le port par défaut du service SSH) pour le port 10010 avec la directive `Port 10010`

Contrôler les logs

Il est important de regarder les logs de temps en temps. Les tentatives de connections infructueuses sont consignées dans `/var/log/auth.log`.

```
grep Invalid /var/log/auth.log
```

Un outil comme **logcheck** peut être utile pour envoyer systématiquement les logs par email.

Astuces

Accès depuis l'extérieur

Rediriger un port (22 par exemple) de la freebox/livebox/whateverbox vers le port ssh de votre machine pour vous connecter chez vous depuis l'extérieur.

Fail2ban

fail2ban est un utilitaire permettant de bannir automatiquement (pendant une durée configurable) les ip ayant échoué plusieurs fois d'affilée à se connecter.

```
apt install fail2ban
```

Serveur de fichier sftp

Pour limiter SSH au partage de fichiers, voir Configuration d'OpenSSH comme serveur SFTP

1)

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

2)

Au moins 30 caractères, majuscules, minuscules, chiffres, symboles, totalement aléatoire et changé régulièrement