Trusty, BROUILLON

Openssh : un serveur ssh

Introduction

Pré-requis

Installation

Installez le paquet **openssh-server**, ou en ligne de commande :

- `sudo apt-get install -y openssh-server`

Configuration

Le fichier de configuration est **/etc/ssh/sshd_config**. On va modifier quelques valeurs.

Paramètre	**Valeur à fixer (exemple)**	**Commentaire**
Port	N'importe lequel. Par exemple 3134.	Changer le port par défaut permet d'éviter de subir les tentatives répétées des scanners de port. En prendre un qui n'est pas utilisé par une autre application.
Protocol	2	Interdit l'utilisation du protocole SSH v1 pour plus de sécurité.
PermitRootLogin	no	Interdit le login via root (il faudra se connecter puis faire su root) par mesure de sécurité.
X11Forwarding	yes	Pour faire de l'affichage graphique déporté.
MaxStartups 10:30:60	Décommenter la ligne (enlever le #)	Après 10 échecs de connexion, il y a 30% de chances pour qu'une connexion suivante soit bloquée. Ce pourcentage augmente linéairement jusqu'à 100% (100% pour 60 connexions).
Banner /etc/issue.net	Décommenter	(Optionnel) Message d'accueil (éditer /etc/issue.net pour le modifier).
AllowUsers (ligne à ajouter)	Nom de votre compte (Alice)	Nom des comptes autorisés à se connecter en ssh. Vous seuls serez autorisés à vous connecter.
ClientAliveInterval	300	Idle timeout en secondes (300s = 5min).
ClientAliveCountMax	0	Avec la ligne précédente, déconnecte après 5min d'inactivité.
IgnoreRhosts	yes	Désactive un accès peu sécurisé
HostbasedAuthentication	no	idem

Paramètre	Valeur à fixer (exemple)	Commentaire
PermitEmptyPasswords	no	idem, interdit les mots de passe vides.

Utilisation

Connexion

Pour vous connecter, lancez :

- `ssh -p le_port_choisi utilisateur@adresse_ip_de_votre_serveur`

ou utilisez Putty sous Windows.

Le mot de passe à fournir est celui de votre compte utilisateur sur le serveur.

Pour **ssh**, l'argument pour spécifier le port est **-p** (en minuscules) tandis que pour **scp**, c'est **-P** (en majuscule).

(Facultatif) Connexion via clé publique / privée au lieu d'un mot de passe

L'authentification par clé évite de devoir taper un mot de passe à chaque connexion et renforce la sécurité.

Ne désactivez pas l'authentification par mot de passe avant d'avoir généré votre clé SSH et de l'avoir testée, si vous n'avez pas d'accès physique à la machine.

Vous n'auriez plus aucun accès à la machine...

Pour mettre en place l'authentification par clé publique/privée, modifiez le fichier **/etc/ssh/sshd_config** comme suit :

Paramètre	Valeur à fixer (exemple)	Commentaire
PubkeyAuthentication	yes	Activer l'authentification par clé.
PasswordAuthentication	no	Authentification par clé uniquement (désactive l'authentification par mot de passe, si vous le souhaitez).
UsePAM	no	Pour ne plus avoir à saisir de mot de passe. Laisser à yes si vous souhaitez conserver l'authentification par mot de passe.

Génération des clés et connexion

Pour obtenir les clés ssh, lancez sur votre machine la commande

- `ssh-keygen -t rsa -b 2048 -C COMMENT`

qui va générer une clé chiffrée en RSA2048 avec un commentaire COMMENT permettant de l'identifier (-C COMMENT est optionnel et, s'il n'est pas spécifié, le commentaire inséré automatiquement sera user@machine).

Une phrase de passe est demandée, la choisir assez forte pour bien protéger la clé (vous devrez vous en souvenir, sinon la clé sera inutilisable, mais vous pourrez déverouiller automatiquement la clé à la connexion, avec le gestionnaire de trousseaux de Gnome, vous évitant de devoir la saisir à chaque connexion).

On obtient deux fichiers **id_rsa** et **id_rsa.pub**, stockés dans /home/utilisateur/.ssh.

Le fichier **id_rsa** est votre clé privée à garder secrète, le fichier **id_rsa.pub** est la clé publique, à donner au serveur.

Pour donner la clé publique au serveur, utilisez la commande **ssh-copy-id** (si vous avez conservé la connexion par mot de passe).

Deux approches différentes sont envisageables vis-à-vis de la gestion des clés SSH.

Pour certains utilisateurs, une clé == une personne, et dans ce cas, vous avez juste à transmettre les fichiers id_rsa *et* id_rsa.pub à chacune des machines avec lesquelles vous souhaitez vous connecter.

Pour un ordinateur portable qui présente des risques de vol, il est plus simple de ne révoquer l'accès qu'à sa clé ssh, en cas de vol (plutôt que de devoir redistribuer les clés SSH à toutes vos machines).

Désinstallation

Voir aussi

- **(en)** siteofficiel
- **(fr)** site

Contributeurs principaux : jamaique.

Basé sur « TitreOriginalDeLArticle » par AuteurOriginal.